Micro Informatique

CPC

ISSN 0296-5689

REVUE DES STANDARDS AMSTRAD

JEU : 3D SNAKE

EDUCATIF : RELIEF

UTILITAIRE : MONITEUR ASSEMBLEUR DESASSEMBLEUR

ESSAI : LES INTERFACES DK'TRONICS



MENSUEL N° 23 - JUIN 1987

AMSTRAD VERS UN MONOPOLE ?

EDITO

Depuis 7 ans que nous sommes dans la presse, nous en avons vu de toutes sortes. De la tentative de rachat en passant par les pressions nombreuses et diverses, nous avons resisté seuls. Soyons honnêtes, pas tout à fait seuls. Notre avocat du Barreau de Rennes n'y est pas étanger.
Nous avons toujours entendu être maîtres de nos jugements. Plutôt disparaître que se coucher sans se battre.

Aujourd'hui, la société Amstrad International nous ordonne de changer l'un de nos titres. En l'occurrence AMSTAR, le journal des jeunes dont le succès est évident aujourd'hui. On y parle de confusion possible que l'on peut faire croire qu'Amstrad participe à la Rédaction et la cautionne (les lecteurs qui suivent CPC depuis ses débuts savent à quoi s'en tenir sur ce sujet !).

On y parle aussi de confusion préjudiciable à la marque. Les lecteurs apprécieront l'argumentation.

Notre confrère Amstrad magazine, qui pourtant a négocié fort cher le titre, risque lui aussi, un jour, de se voir confronter à des problèmes. Un exemple ? Nous avons toujours loué sa maîtrise dans l'organisation des salons Amstrad expo. Il semble que désormais Amstrad France veuille reprendre à son compte cette exposition. "Merci Amstrad magazine. Vous nous avez préparé le terrain, à nous maintenant", telle est peut-être la pensée des dirigeants d'Amstrad. Dans tous les cas, le lecteur remarquera l'élégance du geste et gageons que les exposants seront très largement sélectionnés, si cet état d'esprit n'est pas modifié.

Aussi avons-nous pu apprendre que la société Amstrad souhaitait mettre en place SA propre revue Amstrad-PC. A l'évidence, il faut faire le vide avant !

L'expérience passée montre que les importateurs ou fabricants qui tentèrent de lancer leur propre revue furent peu crédibles, tant sur le contenu que sur la publicité ! Les parutions furent souvent marginales. En agissant ainsi, il n'y a plus de contradicteurs. Seuls certains produits "bien pensants" peuvent être mis en avant.

Ceci représente un immense danger à deux niveaux.

— Celui de l'indépendance de la presse et de l'information déjà parfois sujettes à caution par les lecteurs.
— Amstrad international est anglais et l'on peut craindre que les stratèges d'Outre-Manche préparent des maintenant les ouvertures du Marché commun en 1992.

Pendant ce temps, les Français se battent sur des sujets, certes importants, mais qui ne sont pas les seuls !
Le Marché de l'Europe, c'est demain !
Espérons avec la France

Florence MELLET
Sylvio FAUREZ
Fondateurs







SOMMAIRE

N° 23

Ce numéro comporte un encart de la Société WEKA

CPC est une publication du groupe de presse FAUREZ-MELLET

Directeur de publication
Sylvio FAUREZ
Rédacteurs en chef
Marcel LE JEUNE – Denis BONOMO
Rédaction
Catherine VIARD
Olivier SAOLETTI
Secrétaire de rédaction
Florence MELLET
Directeur de fabrication
Edmond COUDERT
Maquette
Jean-Luc AULNETTE – Patricia MANGIN
Abonnements – Vente au numéro
Catherine FAUREZ
Tél 99 52 98 11
Service ressort – Réseau
Gérard PELLAN
Tél vert 05 48 20 98
Inspection des ventes :
Christian CHOUARD

Photocomposition
SORACOM
Nathalie CHAPPE – Béatrice JEGU
Photogravure couleur
BRETAGNE PHOTOGRAVURE
IMPRESSION :
Presse de Bretagne
Secrétariat-Rédaction
SORACOM Editions
La Haie de Pan
35170 BRUZ
RCS Rennes B319 816 302
Tél 99 52 98 11 +
Télex SORMHZ 741 042 F
Serveur 3615 + MHZ
CCP Rennes 794 17V
Distribution NMPP
Dépôt légal à parution
Code APE 5120

Distribué en Suisse par SEMAPHORE
Tél 022 54 11 95
et en Belgique par COMPUTER MARKET
170, rue Antoine Dansaert
1000 BRUXELLES – tél. 513 53 58

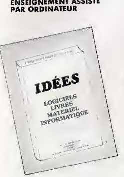
Régie publicitaire
IZARD CREATION
15, rue St-Melaine
35000 RENNES
Tél 99 38 95 33

Chef de publicité
Patrick SIONNEAU
Assistante
Fabienne JAVELAUD

SECURITE INFORMATIQUE

Les entreprises utilisatrices de systèmes informatisés doivent parfois faire face à de lourdes pertes dues à un sinistre informatique d'or dre matériel ou immatériel.
Le Salon Infosec devant se dérouler les 2, 3 et 4 juin prochains réunira 150 exposants qui contribuent, chacun dans leur domaine, à la sécurité des systèmes informatiques. En plus des stands, les personnes intéressées pourront suivre des sessions de formation présentant les méthodes et moyens concrets à mettre en œuvre pour assurer la sécurité d'un ensemble informatique.
Lieu de ce salon INFOSEC 87 : Palais des Congrès, Porte-Maillot, 75017 PARIS

ENSEIGNEMENT ASSISTE PAR ORDINATEUR



Quel est l'avenir de l'informatique à l'école ? Des entreprises en plein essor ont misé sur ce secteur de l'informatique que représente l'EAO (Enseignement Assisté par Ordinateur) et se sont unies pour réaliser un fascicule proposant une sélection de logiciels, de livres, de revues et de matériel informatique pour l'EAO.
IDEES s'adresse tout particulièrement aux écoles, associations et parents d'élèves. Si vous voulez vous le procurer, il suffit de le demander à l'adresse suivante en joignant 2 timbres à 2,20 F pour les frais d'envoi :
Editions SORACOM
La Haie de Pan
35170 BRUZ

AMSTRAD RECOMPENSE

Cette année, Amstrad a été sélectionnée pour recevoir le trophée "Tertiel" et ce pour la performance de son micro ordinateur.
Il faut savoir que le trophée "Tertiel" a pour objectif de mettre en valeur des produits qui sont d'incontestables réussites et Amstrad l'a prouvé en 1986 en devenant le numéro un de l'informatique domestique.

TROPHEE DE L'INFORMATION 1987

Du 7 au 10 mai derniers, s'est déroulé la Coupe de l'Informatique organisée par l'AFIN (Association Française des Informaticiens). Pour cette compétition, 85 bateaux étaient présents parmi lesquels l'équipage d'Omnilogie Citizen qui effectuait là sa première opération de sponsoring. Opération fort réussie puisqu'il a non seulement remporté le trophée de l'informatique 1987, mais également la Coupe du Yacht Club de France pour sa victoire dans le parcours olympique.

SOFTSTRIP COMMUNIQUE

Softstrip ou "La Révolution informatique sur papier" nous indique leur changement de locaux. Voici leur nouvelle adresse :
SOFTSTRIP FRANCE
7, rue de la Gare
92130 Issy-les Moulineaux
Tél. : (1) 40 95 16 66.

AVIS AUX EDITEURS

La société ESAT Software communique aux auteurs et éditeurs de programmes qu'elle tient à leur disposition une série de protections physiques ou logiques sur Amstrad CPC.
Pour plus de renseignements, téléphonez au 56 96 35 23.

PROFESSIONS LIBERALES ET INFORMATIQUE

Vous exercez une profession libérale et vous voudriez adapter un programme de gestion sur votre CPC.
GESTION FINANCIERE est un programme associé à la gestion de cabinet dentaire qui est adaptable à toutes les professions libérales ; de plus, il répond aux conditions spécifiées par les associations de gestion agréées de contrôle des déclarations fiscales.
Pour plus de renseignements contactez Paul Parage-Jonjon au 94 29 69 06.

L'INFORMATIQUE POUR 0 F

Micro & Soft Informatique offre des Ateliers "Open Access" gratuitement aux entrepreneurs qui veulent avoir une gestion performante en attendant d'investir en équipement informatique. Ils peuvent ainsi utiliser des logiciels professionnels de comptabilité, de gestion de stock et facturation, des bases de données, des tableurs...
Pour tout contact :
MSI (tel. 56 94 50 94)

NOUVELLES ASSOCIATIONS

Club informatique
Siège social : Foyer rural de Puyricard — Centre de formation X2000 — Ancienne école — 13540 Puyricard

Club informatique
Siège social : Cité informatique — 9, rue Florent — 69008 Lyon

Club informatique
Siège social : avenue George Sand — 36140 Aigurande

Club micro-informatique de Rivarennes
Siège social : mairie — Rivarennes — 37190 Azay-le-Rideau

Club informatique d'Andernay
Siège social : mairie — Andernay — 55800 Revigny-sur-Ornain

Club neno-ordinateurs
Siège social : 13, rue Thouin — 75005 Paris

Association informatique de Clermont-Dessous
Siège social : Fourtic — mairie — Clermont Dessous — 47130 Port-Sainte-Marie

Club informatique varennois
Siège social : mairie — 03150 Varennes-sur-Allier

Club informatique de l'océan Indien
Siège social : Télémac — 14, rue d'Alsace — 97400 Saint-Denis

Club informatique de Thézy-Glimont
Siège social : mairie — Thézy-Glimont — 80110 Moreuil

Microclub informatique nîmois
Siège social : chez M. Bendinelli (Enc) — 16, rue de Vérone — 30000 Nîmes

ACTUALITÉS

Comme chaque mois, nous vous présentons la liste des nouveaux jeux annoncés pour la gamme Amstrad. Certains seront sans doute commercialisés lorsque vous lirez ces lignes et nous vous en présenterons un banc d'essai dans un prochain numéro d'Amstar ou de CPC.



COKTEL VISION

De l'aventure et de l'action en compagnie de Jolly Jumper et de son fidèle Lucky Luke (ou l'inverse, peut-être ?), NITROGLYCERINE est composé de cinq épisodes d'un style très varié où vous pourrez alternativement avoir votre part d'aventure, votre part d'action, sans oublier votre part de réflexion...

Avec ROBINSON CRUSOE, plongez dans ce que l'on peut appeler un roman animé. Ce style vous permet de vivre l'aventure en prenant les décisions nécessaires à votre survie, en vous immisçant dans les paysages, en relisant des passages de textes. Tout ceci vous permet de prendre connaissance par le jeu de romans classiques.



ERE INFORMATIQUE

L'été arrive bientôt avec les vacances à la clé... Ere informatique y a pensé et a donc décidé de sortir le premier numéro d'une compilation de son catalogue. Elle comprend trois programmes à l'ambiance estivale qui sont : Macadam Bumper, Pacific et Mission 2. Les numéros 2 et 3 sont déjà prévus et sortiront respectivement en septembre et novembre prochains.
CPC : cassette, 150 F ; disquette, 230 F.



ACTIVISION

Avec ENDURORACER affrontez sur votre moto toutes les difficultés du monde ! Vous passerez ainsi des pistes goudronnées à celles très éprouvantes du désert... A noter qu'il est possible de jouer à 2 avec ce logiciel.
CPC : cassette, 95 F ; disquette, 145 F.



CRL

Avec ACADEMY, vous vivez tout simplement la suite de Taucete. Dans ce logiciel, Gal Corp décide de fonder une école de formation pour pilotes d'élite. Si vous voulez devenir un as de l'Académie, vous devrez mener à bien pas moins de 20 missions en combattant à bord d'un Skimmer militaire d'avant-garde...

FIL

Il faut croire que l'arrivée de la fin du premier semestre favorise l'apparition des compilations puisque FIL nous en propose une à son tour avec Pack Amstrad qui regroupe les 4 logiciels suivants : Revolution, The Great Escape, Cauldron II et Sorcery.
CPC : cassette, 149 F ; disquette, 195 F.



ENGLISH SOFTWARE

Avec LEVIATHAN, vous avez entre les mains un jeu classique dans l'espace où il est aussi important d'avoir un vaisseau de maniement très souple que de réussir à survivre. Vous devez naviguer dans trois zones différentes et affronter dix sortes d'ennemis... Alors, bon courage !



INFOGRAMES

Vous avez intérêt à vous entraîner tout l'été aux longues balades en montagne car, dès septembre, vous essaierez de planter votre fanion, par tous les temps, de jour comme de nuit, sur le sommet de votre choix et ce grâce au logiciel BIVOUAC. Ainsi, vous vous mesurerez aux plus hautes âmes du monde.

TOUT 3 SUR SOUND

Bernard POISOT

LE GENERATEUR DE BRUIT ET QUELQUES VALEURS INTERESSANTES

LE GENERATEUR DE BRUIT

Il existe dans SOUND une dernière option, la periode de bruit, prenant les valeurs 0 à 31 Cela génere un bruit pas tres agreable qui se superpose au son defini précédemment Ce bruit permet essentiellement de creer des effets spè ciaux, en procédant ainsi le son emis est haché par une frequence qui se cal cule en divisant 62500 par la pèriode BASIC de bruit Si la pèriode de bruit est à 0, le bruit est arrêté On peut fixer la periode sonore à 0 pour n entendre que le bruit
Il n'y a qu un genèrateur de bruit qui sera le même pour les trois canaux et sa periode sera définie par le dernier ordre pris en compte

Le programme qui suit donne un exemple de ces effets Il imite une locomotive passant devant vous à ' toute vapeur'' avec un effet de relief stèréo afin de simuler le mouvement de la gauche vers la droite qui ne sera, bien sûr, perçu qu avec une installation stéréo En voici le principe
Le train arrive de la gauche et se rapproche lignes 30 50
L'oreille droite commence aussi à entendre le bruit à partir d'un certain rapprochement, mais moins fort lignes 70-100

Quand le train est devant nous, les deux oreilles recoivent la même intensité sonore le volume est maxi fin de la boucle en 100, début de la boucle en 110 Il s'eloigne ensuite à droite A une certaine distance, l'oreille gauche ne perçoit plus rien boucle 110 140
Le son continue à décroître à droite, jusqu'au silence lignes 150-170

Les paramètres qui déterminent le bruit ont ètè regroupés a la ligne 20 d pour la durée f pour la frequence de base et FNbruit pour le bruit superposé Essayez diverses modifications, mais ne jouez que sur un paramètre à la fois. Vous entendrez l'effet des diverses valeurs Conservez ce programme Il vous servira en sous-programme pour utiliser des effets stéreo dans vos œuvres Supprimez la ligne 20 et renumerotez le en 10000 par exemple Vous déterminerez dans votre programme la valeur des paramètres en fonction des effets voulus

```
10 ' essai sonore stereo (locomotive)
20 d=20:f=4095:DEF FNbruit=i+3
30 FOR i=0 TO 10
40 SOUND 1,f,d,i,,,FNbruit
50 NEXT
60 IF SQ(1)<4 THEN 60
70 FOR i=11 TO 15
80 SOUND 1,f,d,i,,,FNbruit
90 SOUND 4,f,d,3*(i-10),,,FNbruit
100 NEXT
110 FOR I=15 TO 11 STEP-1
120 SOUND 4,f,d,i,,,FNbruit
130 SOUND 1,f,d,3*(i-10),,,FNbruit
140 NEXT
150 FOR I=10 TO 0 STEP-1
160 SOUND 4,f,d,i,,,FNbruit
170 NEXT
```

QUELQUES VALEURS INTERESSANTES

L'envoi du code 7 PRINT CHR$(7) provoque un bip qui vide les canaux sonores, arrétant tout son en cours Cela peut se faire aussi en provoquant une erreur, par example en utilisant la touche DEL en début de ligne ce qui générera le même bip
La pèriode BASIC 142 génère le LA des musiciens, sa frequence est 440 Hz.
La période sonore à 0 permet d'obtenir le bruit seul
Si l'on fixe à 0 la durée de la note, c'est l'enveloppe de volume qui dèterminera la durée du son Si cette duree a une valeur négative, l'enveloppe de volume sera répetee le nombre de fois spécifié Bien que le volume ne puisse prendre que les valeurs 0 à 15, la variation imposée par ENV va de -128 à 127 Ceci aura l'effet suivant si le son dépasse 15, il

revient à 0, de même s'il doit descendre à une valeur négative il remontera à partir de 0

Si la durée du pas dans une section d'ENV est 0, elle vaut 256 soit 2,56 secondes

Un numéro d'enveloppe négatif pour ENT repètera l'enveloppe jusqu'à la fin de la note

L'explication des possibilites du son en BASIC est terminee

Essayez diverses combinaisons que votre CP peut engendrer, d'abord simplement sans ENV ni ENT et complétez petit à petit

Vous entendrez de tout bruits et sons, agreables et affreux Il ne tient plus qu'à vous et à votre patience d'éveiller le Jean Michel Jarre qui sommeille en vous

VALEURS AUTORISEES

ETAT DE CANAL certains bits ne pourront avoir simultanément la valeur 1 Ce sont d'une part, les bits 3, 4 et 5 et, d autre part les bits 6 et 7 La valeur mini sera 1, la valeur maxi sera 167
PERIODE SONORE 0 à 4095
DUREE 32768 à 32767
VOLUME 0 à 15
NUMERO D ENV 0 a 15
NUMERO D ENT 15 à 15 (0 exclu)
PERIODE DE BRUIT 0 à 31
NOMBRE DE PAS D'ENT 0 à 239
AMPLITUDE DU PAS D'ENT 128 à 127
DUREE DU PAS D'ENT 0 a 255
NOMBRE DE PAS D ENV 0 à 127
AMPLITUDE DU PAS D ENV 128 à 127
DUREE DU PAS D ENV 0 à 255
ENV et ENT peuvent contenir 5 sec[illegible] chacun[illegible]

AGENDOR

DE QUARTZ SYSTEM

Naël LAGNEU

AGENDOR est la dernière nouveauté de QUARTZ SYSTEM, auteur et éditeur par ailleurs de programmes utilitaires dont nous avons souvent parlé dans ces lignes (les applications graphiques CRISTAL, les gestions de fichiers SAPHIR, la tenue de comptes RUBIS...).

Ce nouveau programme pour PCW est d'une veine tout à fait inédite : il est capable de fonctionner en arrière plan, comme certains utilitaires pour PC tels que SIDEKICK de BORLAND. Expliquons-nous : le programme est chargé dans une zone mémoire qui est ensuite protégée contre l'écriture par une astuce software. Le lancement est obtenu par une combinaison des touches du clavier. Pour SIDEKICK par exemple, il faut appuyer sur les deux touches SHIFT latérales. Pour AGENDOR, la combinaison de lancement choisie est celle obtenue par appui sur ALT, SHIFT et > simultanément. En sortant de ce programme, le système d'exploitation reprend la main. On peut alors naturellement relancer l'application par la même manœuvre ou lancer n'importe quel autre programme. Ce qui est tout à fait impressionnant est que, dans le contexte obtenu, il est encore possible de lancer l'utilitaire en arrière-plan, par la même combinaison de touches, et également de revenir à l'autre programme lancé sous CP/M sans modifier son bon fonctionnement ultérieur, ce qui est tout à fait extraordinaire !... Dommage que LOCOSCRIPT soit exclu du lot, du fait de ses caractéristiques non CP/M...

Le second grand attrait de AGENDOR est son sujet : il regroupe cinq utilitaires très puissants (à l'instar de SIDEKICK par exemple). Son caractère arrière-plan, c'est-à-dire son accès à travers n'importe quel autre programme, n'en prend que plus d'intérêt encore. On obtient ainsi une calculatrice, un bloc note, un calendrier perpétuel, une gestion de fichier et un agenda (qui a donné son nom au programme complet ?...) dont nous allons décrire quelques caractéristiques.

LANCEMENT DU PROGRAMME

Comme nous l'avons indiqué précédemment, le programme est chargé en mémoire (en tapant simplement AGENDOR sous CP/M), puis lancé par appui simultané sur ALT, SHIFT et >. Au bout de quelques secondes, l'écran se brouille uniformément et on a le choix entre deux possibilités : OUVRIR, c'est à dire entrer dans AGENDOR, ou FIN (quitter le programme). Une fois notre utilitaire ouvert, un menu principal propose le choix entre les 5 applications (FICHIER, CALCULATRICE, CALENDRIER, BLOC NOTE, AGENDA), l'option voir que nous décrirons ensuite et la fermeture des fichiers pour quitter AGENDOR.

FICHIER

Cette option étant choisie, la figure 1 nous montre comment se présente l'écran de notre PCW : la fenêtre attribuée à la gestion de fichier apparaît en clair, le reste de l'écran restant brouillé (hormis la petite zone détaillant les commandes).
Chaque fiche se présente sous forme de 7 rubriques (société, nom, adresse 1, adresse 2, ville, code postal, téléphone) et d'une zone commentaire de 7 lignes de 20 caractères (soit 500 fiches sur une face de disquette du lecteur A, ou 2000 fiches sur une disquette double densité). Toutes les fonctions classiques d'une gestion de fichiers sont assurées de manière parfaite. L'utilisation des touches spéciales du PCW est très judicieuse et l'ergonomie générale du programme ainsi que sa facilité d'utilisation sont tout à fait remarquables.

Visualisation

Les fiches défilent à l'écran par action sur les touches + et - encadrant la barre d'espacement. La touche FIND permet d'entrer un nom de société qui nous conduira directement à la fiche cherchée ou qui nous permettra d'en créer une nouvelle. La touche CUT permet d'effacer la fiche courante et on peut l'imprimer directement par appui sur RELAY.

Recherche et tri

Les touches PASTE et COPY permettent de choisir certaines fiches, la première en spécifiant pour les rubriques choisies des valeurs mini et maxi, la seconde en spécifiant des mots-clés dans les zones voulues. Cette façon de procéder couvre la totalité des problèmes courants de tri et de recherche. La sortie des résultats peut se faire de différentes façons :
- A l'écran, fiche par fiche.
- Sur des étiquettes, en paramétrant leur taille et leur nombre.
- Sous forme de listes papier.
- Sur un fichier disque sous format ASCII pour liaison avec tout autre progiciel classique.

CALCULATRICE

Le choix de cette option fait apparaître clairement la calculette dans sa fenêtre. Son emploi est des plus simples : appuis sur les chiffres voulus au clavier et chaînage possible des quatre opérations de base +, -, × et / Cet outil est très utile en dépit de son apparente futilité.

CALENDRIER

Il s'agit d'un calendrier perpétuel de 1900 à 1999. Les touches + et - incrémentent et décrémentent le mois ou l'année concernée. La fenêtre réservée indique alors la correspondance jour de la semaine date journalière.

BLOC-NOTE

Il s'agit d'un carnet dont chaque page peut contenir 10 lignes de 30 caractères (583 pages en A, 2334 pages en B double intensité). On peut feuilleter les pages (touches + et -) et les imprimer (figure 2). Une caractéristique très intéressante de ce bloc-note est de pouvoir chercher un groupe de lettres ou un mot-clé à travers toutes les pages du carnet et retrouver ainsi rapidement un fait donné ou établir des corrélations entre pages.

AGENDA

La taille d'une page AGENDA est sensiblement la même que celle d'une feuille du bloc note. On sélectionne d'abord le jour, le mois et l'année à l'aide des touches du curseur et des touches + et -. Dans une page donnée, l'utilisateur entre ses différents rendez vous. On peut alors imprimer la journée (figure 3), la supprimer ou changer de jour. La recherche par mot-clé permet de retrouver une journée donnée à partir du simple nom d'une personne à rencontrer par exemple !...

VOIR

Cette dernière option est illustrée figure 4. Toutes les fenêtres sont alors clairement visibles et l'écran n'est plus du tout hachuré.

On peut beaucoup vanter ce logiciel, à tous les niveaux :

— Son caractère novateur et inédit le rend très attrayant.

— Les utilitaires proposés sont très complets et très bien réalisés.

— L'ergonomie du programme est tout à fait remarquable (utilisation des touches particulières, simplicité des commandes, clarté de la notice).

Nous sommes convaincus qu'il fera un tabac sur les PCW et puissent les concepteurs et les éditeurs s'aligner sur les qualités de ce produit !

Figure 1
L'écran principal d'AGENDOR, avec la gestion de fichiers activée

Figure 2
Une page de bloc-note

```
---------------------------------Page        1
Ceci est la première page de
mon bloc note  L'aspect le
plus intéressant est l'option
de recherche à travers toutes
les pages d'un groupe de mots
ou de mots clés
```

Figure 3
Une page d'agenda

```
JEUDI    14  MAI        1987
08 h 00 Rendez vous dentiste
09 h 15 Téléphoner à DUPONT
10 h 00 Visite de DURAND (Pressing)
12 h 00 Invitation repas municipalité
15 h 00 Régler compte MARTIN
16 h 30 Aller chercher Fabien école
18 h 00 Téléphoner MEDIASERVICE (Fons)
   h
   h
   h
```

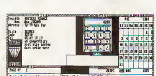


Figure 4
AGENDOR nous montre tous ses visages !...

LES FRACTALES EN LOGO

Alain PRE

De plus en plus de jeux (d'arcades ou d'aventures) utilisent un générateur fractal pour créer un décor sans cesse différent : le joueur a l'impression d'évoluer dans un paysage infini. Dans le cas de ces jeux, il s'agit de fractales en trois dimensions (relief). Nous vous proposons de découvrir les fractales à deux dimensions qui sont bien sûr beaucoup plus faciles à programmer.
Nous avons utilisé Logo car il est idéal pour réaliser ce genre d'application. Si vous ne connaissez pas encore ce langage, voilà peut être l'occasion de vous y mettre.

COMMENT CREER UNE COURBE FRACTALE ?

Pour construire une fractale, il faut tout d'abord choisir un "générateur" ou "motif de base" (quelque chose de très simple comme, par exemple, le dessin de la figure 1).

Ensuite, il faut trouver un moyen pour assembler plusieurs fois cette forme (réduite en dimension) tout en respectant la géométrie du motif de départ.
Comment ! Ce n'est pas clair ? Alors reprenons notre exemple : nous allons remplacer chacun des segments de droite par une copie (à l'échelle 1/3) du motif de base. Ce qui va donner le dessin de la figure 2.

Et ainsi de suite : nous remplaçons encore chaque nouveau segment par le motif de base réduit neuf fois jusqu'à obtenir le dessin de la figure 3.

Théoriquement, il faudrait poursuivre ce travail jusqu'à l'infini. On obtiendrait alors quelque chose comme le dessin n° 4.

Mais, comme le temps c'est de l'argent, nous allons nous arrêter bien avant l'infini. De toutes façons, il faut rester dans les limites imposées par la définition de l'écran.

PROGRAMMATION EN LOGO

Là, c'est vraiment très simple : nous allons d'abord (pour fixer les idées) créer une procédure qui dessine uniquement le motif de base. Nous allons même lui donner un paramètre, ce qui permettra de dessiner des motifs de base de n'importe quelle dimension (voir listing 1 et figure 6).

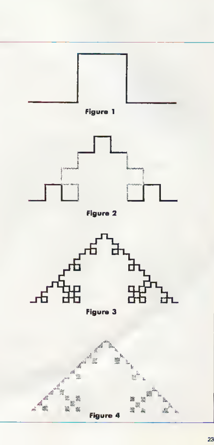


Figure 1

Figure 2

Figure 3

Figure 4

Listing 1



```
to motif :taille
fd :taille
lt 90
fd :taille
rt 90
fd :taille
rt 90
fd :taille
lt 90
fd :taille
end
```

Si vous voulez tester cette procédure, vous pouvez taper par exemple : **motif 20**

Pour transformer ce dessin en courbe fractale, il suffit d'une légère modification de la procédure : remplacer tous les traits (**fd :taille**) par un motif plus petit. Vous savez, bien sûr, modifier une procédure ; il faut taper : **ed [motif]**. Vous obtiendrez le listing 2.

Listing 2

```
to motif :taille
if :taille < 3 [fd :taille stop]
motif :taille / 3
lt 90
motif :taille / 3
rt 90
motif :taille / 3
rt 90
motif :taille / 3
lt 90
motif :taille / 3
end
```

Comme vous l'avez sans doute remarqué, nous avons ajouté une ligne d'arrêt (if **:taille** < 3 ...). Ceci, bien sûr, pour éviter à LOGO d'explorer les profondeurs de l'infini. Vous pouvez modifier à volonté cette ligne d'arrêt pour changer la définition de la courbe :

if :taille < 10 [fd :taille stop]
(peu de définition)
if :taille < 1 [fd :taille stop]
(définition maximale)

Vous voulez voir ce que cela donne ? Alors tapez **motif 30**.
La courbe n'est pas bien centrée sur l'écran. Ce n'est pas bien grave : il suffit, pour arranger cela, de créer une procédure supplémentaire qui initialisera l'écran et la position de départ de la courbe (listing 3).

Maintenant, tapez simplement **départ** pour avoir une courbe fractale correctement positionnée sur l'écran.
Reconnaissez que, pour obtenir un dessin aussi complexe en deux procédures et moins de vingt lignes de programme, il faut vraiment s'appeler LOGO.

Figure 5

Listing 3

```
to depart
cs
pu
setpos [-120 0]
seth 90
pd
motif 81
end
```

PETITE VARIANTE

Le programme précédent dessine la courbe fractale tout de suite avec la définition que vous avez demandée. Il peut être intéressant de faire dessiner toutes les étapes, en partant de la définition la moins bonne jusqu'à la définition la meilleure. Nous ajoutons un paramètre à notre procédure **motif** (listing 4). Paramètre qui sera utilisé dans le calcul de la condition d'arrêt.

Listing 4

```
to motif :taille :arret
if :taille < :arret [fd :taille stop]
motif :taille / 3
lt 90
motif :taille / 3
rt 90
motif :taille / 3
rt 90
motif :taille / 3
lt 90
motif :taille / 3
end
```

Ensuite, nous créons une procédure **étape** (listing 5) étape qui appellera **motif** avec des valeurs d'**:arrêt** de plus en plus petites.

Listing 5

```
to etape :arret
if :arret < 1 [stop]
motif 81 :arret
etape :arret / 3
end
```

Et, enfin, nous modifions **départ** pour appeler non plus **motif** mais **étape** (listing 6).

Listing 6

```
to depart
cs pu
setpos [-120 0]
seth 90
pd
etape 81
end
```

Pendant que nous y sommes, voici encore une petite amélioration : plutôt que de faire superposer les courbes, nous allons effacer le tracé de l'étape précédente avant de redessiner. Mais pour que le résultat soit impressionnant, il faut effacer la courbe au fur et à mesure de l'avancement du nouveau tracé. La procédure **motif** prend alors sa forme finale telle que le montre le listing 7.

Listing 7

```
to motif :taille :arret
if :taille < :arret [fd :taille stop]
pe fd :taille bk :taille pd
motif :taille / 3
lt 90
pe fd :taille bd :taille pd
motif :taille / 3
rt 90
pe fd :taille bk :taille pd
motif :taille / 3
rt 90
pe fd :taille bk :taille pd
motif :taille / 3
lt 90
pe fd :taille bk :taille pd
Motif :taille / 3
end
```

D'AUTRES COURBES...

Maintenant que vous savez comment inventer et programmer une courbe fractale, vous n'allez pas vous en priver. Les listings 8 et 9 vous montrent quelques exemples.



Listing 8



```
to arbre
cs
ht
branche 64
end

to branche :long
if :long < 3 [stop]
fd :long
rt 45
branche :long / 2
lt 90
branche :long / 2
rt 45
bk :long
end
```

Il est interessant de faire varier l'angle entre les branches et de désymétriser l'arbre (une branche avec :long / 2 et l'autre :long / 3 par exemple).

Tapez **exodus 60** et vous allez voir ce que vous allez voir.
Il y a des ouvrages entiers consacrés aux courbes fractales. Autrement dit, il y en a beaucoup d'autres sortes. Mais il est plus amusant de créer les siennes. Essayez vous verrez.

Listing 9

```
to exodus :taille
if :taille < 3 [stop]
repeat 4 [fd :taille exodus :taille
/ 3 rt 90]
end
```

LEXIQUE DES PRIMITIVES UTILISEES

cs (clear screen)
Passe en mode graphique et efface l'écran.

end
Signale la fin d'une procédure.

fd x (forward)
Trace un segment de longueur x à partir du point courant, dans la direction courante.

ht (hide turtle)
Ne fait plus afficher la pointe de flèche, ce qui augmente la vitesse de tracé.

lt x (left)
Change la direction courante de x degrés, sur la gauche.

pd (pen down)
Tous les déplacements du point courant produiront un tracé.

pe (pen erase)
Efface au lieu de tracer.

pu (pen up)
Lève le crayon, c'est-à-dire autorise les déplacements du point courant sans tracé.

rt x (right)
Change la direction courante de x degrés, sur la droite.

seth x
Fixe la direction courante à x (degrés).

setpc x
Change la couleur des tracés (x : 0 à 3).

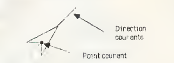
setpos [x y]
Fixe la position du point courant à (x, y).

st (show turtle)
Affiche la pointe de flèche.

to
Annonce la définition d'une procédure.

RAPPEL DE GEOMETRIE LOGO

En LOGO, tous les tracés se font à partir du point courant et dans la direction courante. Ces deux paramètres sont représentés sur l'écran par une pointe de flèche.



Il est possible de changer la direction courante (rt, lt, seth) et le point courant (fd, bk, setpos) en produisant un tracé (pd) ou sans laisser de tracé (pu).
Certains "simples d'esprit" ont assimilé cette représentation mathématique aux évolutions d'une tortue. Animal qui se serait réincarné dans la pointe de flèche précédemment citée.
Soyons indulgent envers ces personnes et laissons les martyriser ce qu'elles croient être un sympathique quadrupède carapaçonné.

BIBLIOGRAPHIE

LOGOMONDE
(Hatier)
Livre + cassette Thomson, Apple. Une trentaine d activités LOGO abondamment commentées (A. Pré. G. Godimier, M. Bonneton).

LOGOMONDE 2
(Hatier)
Livre + cassette Thomson. Intelligence artificielle et LOGO (A. Pré, G. Godimier)

LOGOMONDE AMSTRAD
(en cours d'édition)
Livre + disquette Amstrad. Des exemples de ce qu'il est possible de faire avec DR LOGO sur AMSTRAD (A. Pré, G. Godimier).

GERESCO
(Hatier)
Logiciel de gestion PC ou compatible. Gestion des élèves, analyse de résultats, éditions de bulletins (A. Pré)

LE MONSTRE DE L'INFO A L'ECOLE
Revue Diffusion. Ademir. L'informatique dans l enseignement (G. Godimier)

LES 3 JOURS AMSTRAD, PC, MINITEL DE LYON

Olivier SAOLETTI

Les 15, 16 et 17 mai, s'est déroulé à Lyon une exposition concernant les matériels AMSTRAD, le minitel et les compatibles IBM. Heureux abonnés et lecteurs de CPC, AMSTAR ou PCompatibles, une réduction vous était offerte sur le prix d'entrée. Ceci a dû sans doute vous inciter à faire le déplacement puisque le nombre d'entrées a augmenté de 500 par rapport à la 1ère édition du salon. C'est donc 2500 personnes qui ont arpenté les allées où éditeurs et revendeurs présentaient leurs produits.

Succès certain pour Sémaphore avec le scanner de Dart et Graphpad II, une tablette graphique pour PCW munie d'un logiciel permettant toutes sortes de manipulation d'objets définis en mémoire. Toujours sur PCW, la série des "8000" s'est enrichie d'un MASTERFILE. Il s'agit d'une gestion de fichiers sophistiquée qui utilise au mieux les capacités du disque virtuel.

Amstrad ayant sorti un compatible (vous ne le saviez pas ?), Tasword et Tasprint, deux logiciels connus, se sont adaptés aux nouvelles machines, le tout pour un rapport qualité/prix des plus satisfaisants.

Autre pôle d'attraction, le stand Wings avec le logiciel AMX Pagemaker tournant sur 464 plus une extension mémoire de 64 Ko. La P.A.O. (Publication assistée par ordinateur) étant actuellement en vogue, les CPC ne pouvaient rester en arrière... Muni de la souris AMX, (elle n'est pas obligatoire, mais elle offre, outre la facilité d'utilisation, un "look" sensationnel), Pagemaker permet, à coup de menus déroulants et d'icônes, de fabriquer un petit journal. Il est même possible d'intégrer des images digitalisées et de les disposer à l'intérieur de vos textes.


Gérard Peilon (à gauche) et Michel Archambault en pleine action sur le stand CPC.

Le digitaliseur ARA de Jagot et Léon faisait une démonstration de ses possibilités en se connectant sur un téléviseur. Même débauche d'images avec le logiciel Pastel qui permet la composition de pages VIDEOTEX avec quelques options supplémentaires telles que la transformation des images digitalisées par ARA et la possibilité de fabriquer fenêtres et masques. Le dernier maillon de la chaîne est un serveur 4 voies possédant une extension mémoire de 255 Ko (celle-ci permet un stockage et un traitement des images très rapide).

Du côté purement logiciel, on pouvait trouver Orthogus, un programme éducatif sous forme d'enquête policière (CFI), ainsi qu'un utilitaire mathématique traitant des fonctions numériques. Logys, outre Maths collège et Maths-école déjà testés dans CPC, présentait plusieurs programmes à orientation professionnelle : Financius, gestion d'association, analyse financière...

Sur PC, Micro Star's proposait GT+, un logiciel de gestion intégrée comprenant gestion de stock, comptabilité, gestion commerciale, courrier électronique, avec des possibilités d'extension comme une connexion caisse enregistreuse ou une comptabilité analytique. Autre programme présenté par Micro Star's, Distrib, un gestionnaire pour magasin de prêt-à-porter. Le club C.I.A. (Club Informatique Amstrad) tenait un stand bruyant puisqu'envahi d'exclamations, de sonorités stéréophoniques et amstradiennes.

Autre club présent par l'intermédiaire de quelques membres : Ams'cercle dont un des animateurs, Michel Archambault est venu nous rendre visite sur le stand CPC. Cette exposition a été pour nous l'occasion de répondre à vos questions et pour vous la possibilité de vous procurer les anciens numéros (pas tous malheureusement !).

C'était également le moment de se procurer des produits à tarif réduit tels que disquettes 3 et 5 pouces (Majuscule et France-disquette) ou quelques unités centrales (on pouvait même trouver des ATARI 520 ST sur lesquels tournaient de fabuleuses "démos"). Si vos moyens financiers étaient plus importants, MEI pouvait combler vos désirs avec toute une gamme de fournitures pour l'informatique d'entreprise.

Comme vous le constatez, il y en avait pour tous les goûts, le tout dans une ambiance de grand club informatique.



Le digitaliseur ARA connecté à un 6128

Une démonstration magistrale de l'utilisation de la tablette Graphpad II sur PCW.

Le stand Majuscule, ses PC 1512 et ses disquettes 3 pouces à prix "cassés".

Les interfaces



LE SYNTHETISEUR VOCAL

Faire parler l'ordinateur est toujours amusant . les compte-à-rebours sont plus réalistes, lors du tir d'un missile , les desiderata d'un robot, exprimés en clair, peuvent apporter une touche d'humour à votre dernier programme. Seule l'imagination limitera les possibles utilisations d'un synthétiseur vocal .

Construit autour du célèbre SPO 256, l'un des circuits intégrés "vocaux" les plus anciens, le synthé OK'Tronics a pour seul inconvènient son accent britannique très prononcé. Avec lui, il sera difficile de parler en bon français . Essayez de lui faire prononcer la lettre "U" par exemple ! Ceci ètant dit, tout le reste est fort bien pensè.

L'utilisateur apprendra avec plaisir que ce synthétiseur est doté d'un amplificateur stéréo. Cet ampli viendra se substituer avantageusement à celui qui équipe l'AMSTRAO. Désormais, toutes les illustrations sonores (vocales ou musicales) seront reproduites sur les 2 haut-parleurs livrés avec l'interface. Oeux réglages sont accessibles sur l'amplificateur : le volume sonore et la balance droite-gauche (ou gauche-droite si vous êtes un lecteur susceptible et engagé politiquement). Le second réglage s'effectuera à l'aide d'un petit tournevis , il a déjà été pré ajustè en usine.

Le synthétiseur DK'Tronics est commercialisè en plusieurs versions, fonction de la machine qui doit l'accueillir et selon que le logiciel est livrè sur support magnétique ou en ROM.

C'est cette dernière version (logiciel en ROM) que nous avons testèe. Sentant grandir votre impatience, je ne vous ferai pas attendre davantage.

Le CPC reconnaît la présence de la ROM du synthé (je ne vais pas ècrire le mot en entier à chaque fois. .) en affichant "SPEECH ROM VER. 1.1". L'instruction ISPEAK permet d'initialiser la ROM et lance un test interne donnant tout à coup la parole à l'AMSTRAO. La liste des commandes disponibles est affichée : il ne reste plus qu'à les essayer !

Oisposant d'un espace limité pour ce banc-d'essai (on ne va tout de même pas dètailler la notice !), nous allons voir qu'il existe, en fait, deux grands modes de fonctionnement, les autres en dérivant plus ou moins.

A chaque phonéme est associé un code (il y a ainsi 59 sons et 5 durèes de pause) dècimal qui, envoyè à l'adresse du synthè, produit un son. Ceci ètant dit, on peut donc programmer "son par son", par l'intermédiaire de listes de OATA. Les octets lus sont envoyès dans un buffer

DK'TRONICS

Amstrad construit des ordinateurs ; DK'Tronics les complète avantageusement par des interfaces comblant certaines lacunes de la machine. Présentées en boîtier plastique de belle facture (mais non, pas le prix, la réalisation !), les extensions peuvent être accouplées, non pas pour faire des petits, mais pour ouvrir à l'utilisateur des horizons nouveaux. Curieusement, DK'Tronics édite un manuel d'utilisation commun à toutes les interfaces. L'acheteur d'un synthétiseur vocal se laissera-t-il tenter, au vu des possibilités décrites dans le manuel, par un crayon optique ou un RAM-Disque ?
Les périphériques DK'Tronics que nous avons testés pour vous sont tous disponibles chez ORDIVIDUEL. En vous recommandant de "CPC", le meilleur accueil vous sera réservé.



qui pourra en contenir jusqu'à 250. Ces sons peuvent être envoyés en mode "direct" ou en mode "programme", grâce à un astucieux contrôle des interruptions, le synthé ne ralentira pas trop votre programme. Grâce aux sons élémentaires, on pourra former des mots, mais aussi produire des effets sonores assez recherchés.
L'autre mode de fonctionnement fait appel à un convertisseur Texte/Parole. L'idée est excellente car on peut ainsi introduire du texte en anglais, sous forme de chaînes de caractères, et le faire prononcer au St T (j'en ai marre d'écrire synthé ...) au moyen d'un simple PRINT.
Le logiciel est conçu pour reconnaître certaines règles d'exception de la langue anglaise, ce qui simplifie bien notre tâche.
J'ai bien dit "langue anglaise" car, pour le français, il faudra ruser. Certains mots anglais devront être orthographiés différemment (par exemple, "final" est prononcé presque "à la française" et devra être écrit "fianol" pour ressembler à de l'anglais). Suffit de savoir, quoi ! La lecture de la notice nous apprend que le buffer de texte peut contenir jusqu'à 100 caractères, ce qui permet de former des phrases suffisamment longues. Avec sa contenance de 250 phonèmes, le buffer de parole fera parler la machine pendant 45 secondes... mais une commande appropriée permet de lui couper le sifflet !
Profitant d'une autre caractéristique de ce St T (j'en profite !), l'AMSTRAD pourra dicter les listings ! En effet, tout ce qui apparaît à l'écran sera énoncé par la machine. Hélas, ceci restera parfois assez confus, la voix robotisée s'accommodant mal de certaines associations de sons.
Le synthétiseur (quel courage !) DK'Tronics est également programmable à partir de l'assembleur. La notice donne quelques explications à ce sujet. Cette notice se termine par une table des phonèmes, listés par numéro d'ordre avec, pour chacun d'eux, un exemple de mots extraits du vocabulaire anglais.
Malgré quelques petits défauts (notamment une voix un peu trop métallique lors de l'utilisation du convertisseur texte/parole), le saint thé (normal, il est british...) DK'Tronics est séduisant par ses possibilités. Le constructeur offre à l'utilisateur, en plus du synthétiseur (c'est la dernière fois que je l'écris), un ampli stéréo et ses 2 HP... pour le même prix !



LE RAM-DISQUE

Un disque de 2S4 Koctets utilisables, voilà ce qui se cache sous le nom de "Silicon Disc" RAM-Disque en "français" (je vois les habits verts de nos académiciens en frémir !)

A quoi peut bien servir un RAM-Disque ? A charger et sauvegarder des programmes, fichiers, blocs de données, beaucoup plus rapidement. Et ça marche sur n'importe quel CPC ? Oui mon bon Monsieur.. à condition que ledit CPC soit déjà équipé d'au moins un lecteur de disquettes. Le RAM-Disque se comporte alors comme une disquette normale aux différences suivantes près :

— Les fichiers sont lus et écrits beaucoup plus rapidement.

— Ils sont définitivement perdus en cas de coupure de courant.

— On les retrouvera même après un RESET complet.

Quel intérêt peut-on avoir à utiliser un RAM-Disque ? Le gain de vitesse lors de la manipulation de gros fichiers ..

Le RAM-Disque DK'Tronics est utilisable sous AMSDOS (en BASIC) ou sous CP/M (2.2 ou CP/M+). L'interface pour 464 et 664 se présente en 2 boîtiers : l'un contient le système d'exploitation, l'autre la RAM. L'un dans l'autre.. ils se connectent (aïe, pourvu que M. Pasqua ne fasse pas interdire CPC !) et sont unis pour le meilleur (et pour le pire, si vous les utilisez mal) et pour le grand bien de l'ordinateur hôte.

Selon que votre CPC est déjà doté d'une ou 2 unités de disquettes, le RAM-Disque deviendra le disque B ou C.

Après une initialisation relativement simple, destinée à indiquer au système la présence d'un disque supplémentaire, l'utilisateur transférera ses fichiers de la disquette vers le RAM-Disque. Sous CP/M, la procédure de configuration du Setup est décrite dans le manuel de DK'Tronics, ce qui ne laissera pas errer inutilement l'utilisateur. Toujours sous CP/M, il faudra utiliser PIP pour transférer les fichiers car COPYDISC et DISCOPY, plus gourmands en TPA, ne tournent pas.

Les temps (sont durs..) comparés de chargement d'un fichier sont éloquents : pour 30 K de BASIC, il faut 7 secondes sur disquette et moins de 1,5 s avec le RAM-Disque.. Spectaculaire également, le chargement d'un écran de 17 K à partir du RAM-Disque ..

Si vous êtes consommateur de gros fichiers, pensez à ce RAM-Disque. Au fait, il existe aussi en version 64 K ■

L'EXTENSION RAM 256 K

Encore une extension de mémoire, mais qu'il ne faut pas confondre avec le Ram-disk. Celle-ci a pour vocation de gonfler votre CPC 464, 864 ou 6128 en lui attribuent 256 Ko supplémentaires.

Cette mémoire est divisée en 16 banques de 16 Ko qui seront accessibles par commutation au moyen d'un jeu de commandes RSX. Mais n'allez pas imaginer, comme le font bien souvent les débutants, que vous allez pouvoir écrire des programmes en BASIC de près de 200 K. Ce n'est pas si simple ! Par contre, elle vous rendra les plus grands services pour sauvegarder tout ce qui est gourmand en mémoire, comme par exemple vos pages d'écran, vos tableaux ou vos chaînes de caractères, allégeant d'autant vos programmes en BASIC.

L'interface est livrée avec une disquette contenant le logiciel d'implantation des RSX. Notez que vous avez le choix de l'adresse d'implantation et que, par défaut, le programme s'installera à l'adresse la plus élevée disponible. Dès que cette opération est effectuée, l'écran vous indique le nombre de banques disponibles. Voyons maintenant le rôle des différentes fonctions complémentaires, mais soulignons au préalable que la notice d'emploi fort détaillée vous facilitera la première prise en main.

ISAVES,n et ILOADS,n permettent de stocker en mémoire et de rappeler des pages écrans, n étant le numéro de bloc. Les anglophiles auront interprété le S de la commande à Screen.

Syntaxe analogue pour ISAVEW,F,n et ILOADW,F,n (W comme window) qui ont respectivement pour fonction de sauvegarder et de rappeler des fenêtres caractérisées par leur numéro, n étant toujours le numéro de bloc. Ces deux commandes vous permettront de réaliser facilement des menus superposés à la Mac-Intosh.

IHigh et ILow chargent deux images en mémoire centrale, l'une à la place habituelle (high) et l'autre immédiatement au-dessous (low), ce qui, grâce à la commande associée ISwap, vous permettra, par commutation de pages, de réaliser des animations graphiques.

Quittons maintenant le domaine de l'écran pour voir le stockage de données qui s'effectue de manière analogue par ISAVED (comme Data) et ILOADD. Les données peuvent être des tableaux, des variables ou des chaînes de caractères.

Heureusement que la notice est très claire à ce sujet et l'expérimentation, à l'aide des petits programmes de démonstration inclus, vous permettra très facilement de tirer le meilleur parti de ces deux commandes.

Il nous reste à citer les quatre dernières commandes qui sont IASKRAM, IPEEK, IPOKE et IBANK qui s'adressent au programmeur plus expérimenté qui pourra également, s'il a bien assimilé la topographie de la mémoire de l'AMSTRAD, écrire ses propres routines en BASIC ou en assembleur pour s'affranchir des RSX d'origine.

Signalons enfin que cette extension permet d'utiliser CP/M+ tel qu'il est livré avec le 6128 et que la commande IEMULATE transformera votre 464 ou 684 en véritable 6128■



LE CRAYON OPTIQUE

Le crayon optique DK'Tronics se connecte par l'intermédiaire d'un mini jack à son boîtier d'interface qui présente, vous vous en doutez, un air de famille avec les autres périphériques de la marque.
Ce boîtier contient, outre l'électronique du stylo, une ROM dans laquelle est implanté un logiciel de dessin exploitant les possibilités de l'outil. La mise en œuvre de l'ensemble est extrêmement simple. Il suffit de taper ILIGHTPEN pour voir apparaître le premier menu du logiciel intégré dans la ROM qui aura détecté la présence éventuelle d'un lecteur de disquette et vous proposera de charger ou de sauvegarder vos dessins sur le support magnétique adéquat. "Next" permet de passer au menu suivant et il y en a cinq en cascade ainsi de suite. Le second vous propose d'effacer l'écran et de calibrer le crayon optique qui a besoin, pour fonctionner correctement, d'un niveau de lumière ni trop faible ni trop élevé. Vous règlerez donc le brillance de votre écran au niveau suffisant pour pouvoir déplacer la petite cible aux quatre coins de l'écran. De temps à autres, cette petite cible a des tremblements erratiques, mais la notice précise que c'est sans importance car le dessin pourra être corrigé à la loupe par la suite. Les quatre flèches de direction vous permettront de centrer exactement la cible sur le stylo.
Une fois cette opération effectuée, vous voilà prêt pour le dessin. Vous disposez pour cela d'une palette de 10 couleurs qui vous serviront pour les crayons (disponibles en 4 épaisseurs), pour l'aérographe, les textes qui peuvent être horizontaux ou verticaux et, enfin, pour la fonction de remplissage de formes, matérialisée par un robinet. Comme tous les produits concurrents, le stylo DK'Tronics a besoin de lumière pour fonctionner, d'où une certaine difficulté pour choisir une couleur sombre. Mais l'auteur du programme y a remédié en autorisant la sélection de couleur par la bordure jaune qui entoure chaque teinte de la palette. En dehors des cercles et des rectangles accessibles par le menu, il est possible de tracer des figures complexes au moyen "d'élastiques" tendus entre des punaises. Votre courbe complexe sera ainsi constituée d'une succession de lignes courtes et cette fonction se révèle très pratique à l'usage. Notons aussi la présence d'une fonction de copie, d'agrandissement ou de réduction d'une partie du dessin. La fonction loupe est ici appelée "calepin". Au moyen d'une fenêtre réglable, vous sélectionnez une partie de l'image avant de la transférer vers le calepin où elle apparaîtra en plein écran et là vous pourrez la modifier au pixel près.
Reste un dernier point : l'impression de vos chef-d'œuvres. Vous disposez pour cela d'un petit programme livré sur une cassette et qui vous offre le choix de l'imprimante entre Epson et Amstrad.
Après notre test, nous regretterons simplement que DK'Tronics n'ait pas installé à l'extrémité de son stylo un petit bouton poussoir manœuvrable du bout de l'index, ce qui éviterait de manipuler la touche Enter de la main gauche. Mis à part ce petit regret, ce stylo optique s'est révélé d'un usage fort agréable et la gamme des outils mis à la disposition de l'artiste permet facilement de dessiner avec une grande précision ■

UN PORT D'ENTREES-SORTIES ECONOMIQUE

Eddy DUTERTRE

Les trois versions de CPC disposent d'origine d'un port d'entrées-sorties facilement utilisable pour les bricoleurs. La sortie imprimante centronic est constituée de huit lignes en sortie (DO à D6 et STROBE) et d'une en entrée (BUSY). Ce port est actif par le BASIC avec les instructions PRINT #B ou LIST #B, mais le résultat obtenu est tributaire du logiciel interne de l'ordinateur. Par contre, grâce à deux instructions particulières du BASIC (OUT et INP), il est possible de se rendre maître de la configuration de ces lignes.

L'important est de savoir que, pour les sorties, l'adresse du port est &EF00 et pour l'entrée &FS32. Ainsi, pour mettre toutes les lignes D0 à 06 et STROBE à un, il faudra faire :

OUT &EF00,&7F ou OUT &EF00,127

La ligne sept (STROBE) est inversée par rapport aux autres car elle passe à un en mettant le bit huit à zéro d'où &7F à la place de &FF (01111111 = &7F 11111111 = &FF)
ligne... 76643210 76543210
Pour une ligne sur deux à un :

OUT &EF00,&2A ou OUT &EF00,42
&2A......00101010 (notez le bit 7 à zéro)
ligne....76543210

Inversement, pour lire la ligne en entrée (BUSY), il faudra faire :

A = INP &FS32:A = A AND 64

Si l'entrée est à un, A aura pour valeur 64 sinon A sera nul.

Comment exploiter ces sorties ?

Une première mise en garde. Il n'est pas possible de commander directement, avec ces sorties, des relais ou autres systèmes tels que voyants. Un petit montage buffer est nécessaire (voir schéma). Pour l'entrée, il ne faudra pas dépasser 5 volts. L'idéal est de disposer d'un contact à la masse (interrupteur ou contact de relais).

```
10 REM EXEMPLE DE CHENILLARD
20 REM ********************
30 REM LA VITESSE S'AJUSTE A LA LIGNE 45 (T)
40 FOR N=0 TO 7
50 OUT &EF00,2^N
60 FOR T=1 TO 400:NEXT T
70 NEXT N
80 GOTO 40
```

Schéma d'un buffer.

b
c
e
4,7K
2N2222
UTILISATION
Diode de protection (1N4007)
+ ALIM. EXTERIEURE 12 Volts max
-
venant de la prise IMP. D0 à D6 ou STROBE
Les masses sont à relier a la broche 14 ou 15 de la prise IMP.

Entrée sur BUSY conseillée

broche 11 (busy)
c
b
10K
e 2N2222
broche 14 (masse)
Entree 12 V max

17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19



OCTODE

Zig et Puce

OCTODE est le nom donné au montage proposé, il s'agit d'un mini clavier à connecter sur la prise joystick. Son originalité réside dans le fait qu'il assure au curseur de francs déplacements obliques. Tout indiqué pour les utilitaires à option joystick et technique menus déroulants, il demeure sociable en usage ludique. Pour réaliser ce montage, les néophytes liront attentivement le texte.

PRINCIPE DE FONCTIONNEMENT

Sur un joystick ordinaire, un mouvement oblique s'obtient en pressant deux boutons de déplacement. Par exemple, pour atteindre le coin supérieur gauche de l'écran, la base de la manette de jeu doit presser en MEME TEMPS les contacts HAUT et GAUCHE. Mais cela ne peut pas fonctionner parfaitement dans tous les cas de figure et alors le curseur dévie soit vers le haut, soit vers la droite.

Le dispositif de la figure 1 assure un déplacement oblique UNIQUEMENT lors d'une action SIMULTANEE sur les contacts sollicités. Il s'agit d'une fonction ET, matérialisée dans notre cas par deux diodes et un poussoir (figure 4). OCTODE apporte quatre directions (ou sens plus exactement) obliques.

Il faudra donc huit diodes et quatre poussoirs à cet effet. Les deux feux sont placés dans la partie supérieure du montage et l'un d'eux (F2), est couplé avec la touche centrale (figure 7).

REALISATION

Outillage à rassembler :
- pinces plates,
- pinces coupantes,
- pinces à dénuder,
- fer à souder, soudure,
- tournevis.

Composants à se procurer :
- 11 boutons poussoirs,
- 8 diodes types 1N4148 (C1 C8),
- connecteur joystick femelle,
- câble 1 mètre, 9 conducteurs,
- circuit imprimé (CI).

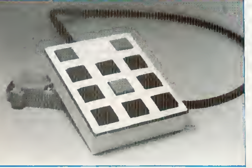
Pour le CI, les néophytes se procureront une pochette KIT DE GRAVURE (mode d'emploi inclus).

PREPARATION

Avant d'implanter les composants, il faut les identifier, repérer leurs connexions, les mettre en forme.

— Le CI : tous les trous sont percés au diamètre de un millimètre (figure 10).
— Le CABLE : comporte neuf fils conducteurs isolés à relier d'un côté au connecteur, de l'autre au CI (figure 7). Les fils seront dénudés (longueurs en mm), étamés et repérés (figures 5 et 9).
— Pour ETAMER un fil, torsader ses brins, chauffer cette extrémité et y déposer un peu de soudure (figure 5).
— Le REPERAGE consiste à associer une couleur à un chiffre pour chaque fil du câble. Exemple : le fil rouge ira d'un côté au 2 du connecteur (figure 8) et de l'autre, bien sûr, au 2 du CI (figure 10).

Et ainsi de suite... Enfiler le capot du connecteur (figure 7) sur le câble afin de ne pas l'oublier.
— Le CONNECTEUR : côté soudure, les broches (ou sifflets) pourront être étamées.
— Les DIODES : plier les pattes. La bague indique la CATHODE (figure 6).
— Les POUSSOIRS : quatre pattes. Rien à faire de particulier.
L'implantation des composants sur le CI se fera telle que le montre la vue de dessus (figure 7) et dans l'ordre suivant : diodes tout d'abord, cathode bien orientée, poussoirs ensuite et, pour finir, le câble, lui-même relié au connecteur suivant repérage (figure 8 à 10).
Soudure : éviter de trop chauffer les dio-



des qui sont relativement sensibles à la chaleur. Couper les pattes (ou queues) après soudure (figure 6).

CONTROLE

Les diodes sont-elles correctement orientées, les fils reliés aux bons endroits, les soudures toutes faites ? Visser le capot sur son connecteur. Coller une plaquette (carton ou matière plastique) sous le circuit imprimé afin d'isoler les soudures. Brancher la prise femelle sur le port joystick de l'ordinateur. Tester OCTODE avec FRUITY-FRANCK ou Z pour les jeux, CHERRY PAINT ou AMSTRADIVARIUS pour les utilitaires.

Les initiés pourront s'attaquer à la version luxe où l'on remarque (figure 2) :

S inverseur des communs 8/9

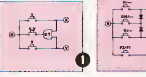
E ON/OFF de la touche centrale

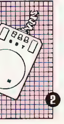
T inverseur des feux 6/7

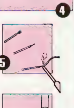
M touches bas-profil disposées en cercle sous une membrane ■

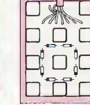


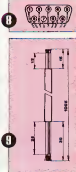














BUG, OU ES-TU ?

Michel ARCHAMBAULT

Un programme nouveau ne fonctionne jamais du premier coup Il y aura toujours au moins un "bug" (ou "bogue") qui provoquera un plantage ou un resultat que l'on n'attendait pas du tout.
Il y a deux sortes de bugs le franc et net, du genre "Syntax error in 340", et puis le farceur, le sournois qui vous fait croire que tout marche bien et qui, de temps a autre, vous affiche un resultat faux Non seulement ce bug fantôme est parfois difficile à remarquer, mais sa localisation va demander beaucoup de ruse et de methode Heureusement il est plus rare
Mais on peut aussi diviser les auteurs de bugs (nous) en deux catégories ceux qui programment et ceux qui recopient des listings publiés dans des revues ou des livres Et là, les causes d'erreurs sont bien spécifiques De ce fait c est la classification que nous allons suivre nous parlerons donc d'abord des erreurs de recopies, puis des erreurs de conception de programme et, enfin, d'un domaine commun, la localisation d'un bug cache

LES ERREURS DE RECOPIE DE LISTINGS

Il serait trop facile d'ouvrir notre parapluie en disant «Bien sûr ! Ils copient sans essayer de comprendre !» car même un bon programmeur a parfois toutes les peines du monde à comprendre un listing ecrit par quelqu un d'autre ! Pourquoi ? Parce qu'ils ont des façons de penser complètement différentes Alors le "copieur" renonce à comprendre la pensee «contre nature» (la sienne) de l'auteur du programme, et il se met à copier «bêtement» en lui faisant confiance C est humain
Premier conseil, bien que cela devrait être une evidence, jetez un coup d oeil au texte et au listing avant de commencer la frappe ! surtout si vous possédez un CPC 464 Peut être y lirez vous qu'il ne s'adresse qu au 6128 !

Les confusions de caractéres

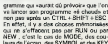
Il est tres facile de mal interpréter certains caractères surtout quand un defaut d'encrage chez l'imprimeur vient compliquer les choses Je pense à tout ce qui ressemble à une barre verticale cela peut-être un I majuscule, un chiffre 1, un L minuscule ou la barre obtenue par SHIFT + a commercial
Nota si vous nous envoyez un programme en vue de publication, ne programmez qu'en majuscules (sauf pour les REM et les textes entre guillemets), et pensez à ne pas utiliser la variable I ou des noms de variables commencent ou se terminant par cette lettre Sinon, attendez vous à recevoir beaucoup de courrier
Autres confusions classiques, le chiffre 0 et la lettre O, 5 et S, le point et la virgule, les deux points et le point virgule, le G minuscule et le chiffre 9 Ajoutons à cela quelques étourderies dyslexiques courantes apres 23 heures 30 du genre slash et anti-slash (les touches sont voisines), < et > M et W en minuscules
Autre cause d'erreurs fréquentes, les espaces (ou blancs) en trop ou en moins lorsque l'on veut indiquer qu'une variable chaîne R$ est "vide" ou nulle, on ecrit R$ = "", c'est a-dire deux guillemets consécutifs Si vous mettez un espace entre eux deux, ça change tout ! R$ n'est plus vide, elle est de un caractere Pour indiquer que R$ = un blanc, il est plus prudent de programmer R$ = SPACE$(1) Autre cause heureusement plus rare certains programmeurs (qu'il faudrait battre !) ont remarque que le blanc est parfois un caractére de séparation en BASIC, alors, dans des lignes de DATA, il mettront un espace en guise de virgule ou dans un ordre PRINT pour remplacer les points-virgules Ça marche mais si le recopieur oublie un de ces blancs, c'est la catastrophe assurée

Les adaptations malheureuses

Pour rendre la recopie moins «bebete», il est humain d apporter çà et là des petites modifications «sans importance», histoire de personnaliser un peu ce que l on tape Tres bien, mais de nombreux pieges vous guettent
Premier cas les noms de variables Afin de rendre le listing plus clair, l auteur a utilisé des noms de variables longs par exemple RAPPORT = A/B, et comme ce nom revient souvent vous décidez que RAP est bien suffisant Oui, mais en poursuivant la frappe le lendemain, voilà que vous l'appelez RAPP Autre cas avec par exemple DIRECTION = 4 trop long ! Vous tapez DI = 4 Et voilà un futur "Syntax error", parce que DI est un mot BASIC, peu usité certes mais c'est un "mot reserve" (voir la liste des "mots reserves" dans votre manuel)
Une autre aventure ? Les numeros de lignes Un programmeur qui tient à conserver son standing fait des sauts de numérotations à des multiples de 100 ou de 1000, à chaque fois qu il aborde une nouvelle etape Ça fait " rangé " et net surtout quand ces numéros de lignes «tout en zéros» sont des REM expliquant ce qui suit
Oui mais il y a le copieur qui se dit que «AUTO c'est pas fait pour les chiens» et aussi que «Les REM ça ne sert à rien, donc je saute » Mais il n'empêche qu il transcrira scrupuleusement IF N = 0 THEN GOTO 800 Des GOTO qui iront se perdre dans les marecages
Vous pouvez vous dispenser du texte de REM de la ligne 2000 mais tapez au moins 2000' sinon gare au "LINE DOES NOT EXIST IN 340"

les impondérables

Là, c'est de la faute de l auteur du pro-



gramme qui «aurait dû prévoir» que l'on va lancer son programme «à chaud» et non pas après un CTRL + SHIFT + ESC. En effet, il y a des choses mémorisées qui ne s'effacent pas par RUN ou par NEW, c'est le cas de MODE, des couleurs de l'écran, des SYMBOL et des KEY DEF. Pour illustrer cela, voici deux gags authentiques.

— Gag n° 1 : ce programme débute par un GOSUB qui définit des caractères par SYMBOL, donc précédés par un SYMBOL AFTER. Premier essai, impeccable ! Tout content, on le relance par RUN. Aussitôt, plantage avec "Improper Argument in 30" ! En 30, il y a le GOSUB qui, lui, débute par SYMBOL AFTER 64 et, comme le précédent, SYMBOL AFTER,64 est resté actif, ça plante. Faire toujours précéder un SYMBOL AFTER par un CALL &BB04 qui élimine les précédents.

— Gag n° 2 : cet utilisateur de CPC couleur détestait programmer en lettres jaunes sur fond bleu foncé. Il commençait toujours une saisie de listing par
INK 1,0 PAPER 2 CLS
donc en lettres noires sur fond bleu ciel. C'est effectivement plus net et reposant. Oui, mais le long programme recopié commençait par PEN2 CLS... Sitôt lancé par RUN écran est vide et bleu ciel, ESC et clavier apparemment inactifs ; affolement ! Reset. Normal avec PAPER et PEN de la même couleur..

Est-ce tout ?

Cette liste de "gaffes" n'est pas exhaustive, hélas. L'auteur s'est inspiré de plusieurs années de «courrier des lecteurs» et d'animation de club informatique, pour recenser les cas les plus classiques ou les plus "curieux". A ce propos, si vous êtes complètement bloqué malgré moult vérifications, n'écrivez pas simplement «J'ai retapé très fidèlement le listing de votre programme TRUC et il ne marche pas. Dites-moi où est l'erreur». Ce ne sera possible que si vous joignez votre listing ou, à défaut d'imprimante, une cassette. Avec un enregistrement le diagnostic sera quasi immédiat, d'où une réponse rapide.

LES FAUTES DE PROGRAMMATION

Comme nous l'avons dit au début, il y a l'erreur "immédiate", genre "Syntax Error" ou "Improper Argument", et l'erreur "fortuite" qui survient un jour dans un programme qui tourne depuis longtemps et ce à la suite d'un concours de circonstances que le programmeur "aurait dû prévoir" (comme pour la fusée Ariane, mais un logiciel est quand même vendu moins cher ...)
Tout cela ne veut pas dire qu'un bug se traduise toujours par un arrêt plantage ; il peut aussi fournir des résultats absurdes qui, pour l'instant, ne provoquent pas de plantages.
Il n'est pas possible de faire une liste de toutes les erreurs possibles en BASIC Ok ? En revanche, il va être bon de dénoncer celles qui sont au Hit Parade, et qui d'ailleurs s'y accrochent... On commencera donc par l'éternel grand champion toutes catégories.

Les initialisations de variables

Contrairement à la plupart des langages, le BASIC initialise toute variable inconnue à zéro. Faites PRINT ZZ, réponse 0, PRINT ABC$, réponse chaîne vide ("").
Alors, on en prend l'habitude et pour démarrer un comptage (ou "incrementation") on écrit simplement K=K+1. Donc, au premier passage, K=0+1 et ainsi de suite. Supposons que K ait atteint la valeur 15. Puis le programme est amené à repasser par là pour de nouveau comptabiliser de 1 à... Si on n'a pas pris la sage précaution de réinitialiser par K=0, le deuxième comptage va démarrer à 16 !
Même remarque avec la classique
WHILE R$="" R$=INKEY$ WEND
au premier passage l'ordinateur attend que vous pressiez une touche. Vous pressez N, donc R$="N", on "passe" le WEND. Deuxième passage par cette ligne : R$ vaut toujours "N", donc "passage" du WEND sans attente d'un INKEY$ au clavier ! Vu ? Il aurait suffit juste avant le WHILE de programmer R$=""
Une autre source d'ennuis est notre manque d'imagination pour les noms de variables, toujours les mêmes lettres qui reviennent, d'où risque d'homonymies catastrophiques. Exemple : vous avez programmé un compteur par I=I+1. Votre programme fonctionne très bien, puis, un jour, vous décidez de le compléter par un nouveau sous-programme qui affiche un message à l'écran. Pour avoir le temps de le lire, vous mettez la classique boucle d'attente de trois secondes
FOR I=1 TO 3000 NEXT
RETURN et le cycle repasse par le compteur qui saute ainsi à 3002. Surprise...
Deux conseils :
— Initialiser vos variables (K=0 R$="") même si cela semble ne servir à rien (pour l'instant...)
— Méfiez vous des homonymes en noms de variables, tout particulièrement lorsque vous retouchez un long programme ancien. Adoptez des noms à consonnances martienne ou venusienne, du genre ZZZ ou WYW, c'est plus sûr que I, N ou K.

Note

Pour les cas épineux, je vous recommande le programme "Références croisées" de TO MINH TRI, publié dans CPC n° 7 page 65 : il liste les variables par ordre alphabétique avec les numéros de lignes où elles figurent. Une pure merveille !

Les farces des valeurs nulles

Diviser un nombre par zéro donne un résultat infiniment grand, un non-sens pour un ordinateur qui réplique par "Division by zéro in..." Donc, à chaque division, posez-vous la question de savoir s'il y a une "chance" pour que le diviseur soit nul. Exemple classique A=B/(C-D). Même si la probabilité pour que C=D est infime (donnerait division par zéro), dites-vous que cela arrivera le jour où vous ferez une démonstration de votre programme. (=loi de MURPHY ou de "l'emm... maximum"). Il faut prendre ses précautions
CD=C-D.IF ABS(CD)<0.001 THEN CD=0.001
A=B/CD
Autre cas, les paramétrages de commandes BASIC, par exemple LOCATE 12,Y PRINT A$
Il est sage de faire précéder par
IF Y<1 THEN Y=1
un risque associé est celui de l'OVERFLOW avec les nombres entiers :
N%=A/B est dangereux. Ecrivez plutôt
IF ABS(A/B)<32767 THEN N%=A/B

Les INPUT et les DIM

Ces causes d'erreurs arrivent en troisième position. Ce sont des sujets assez vastes et nous leur avons déjà consacré récemment deux articles : "La gestion des réponses" dans CPC n° 19, "L'exploitation des tableaux DIM" dans CPC n° 21.

La logique illogique

Les fonctions "logiques" sont les suivantes :
IF... THEN, ELSE, AND, OR, NOT et XOR
XOR(=OU exclusif) est d'un emploi peu fréquent. Ces fonctions ont un fonctionnement TRES STRICT, mais qui ne génèrera jamais de code d'erreur si elles exécutent tout autre chose que ce que vous aviez prévu. C'était à vous de ne pas les employer à tort et à travers. Hélas, c'est facile...
Le risque initial est de traduire en BASIC une pensée en "langue française", exemple «j'accepte les réponses "O" ET "OUI" comme valables», alors le "débutant" programme
IF R$="O" AND R$="OUI" THEN GOSUB 5000
ce qui est "logiquement" impossible, il fallait écrire OR et non pas AND.
Mais la championne des gaffes classiques consiste à mettre plusieurs "clauses" (séparées par des :) sur une même ligne, en oubliant ce principe essentiel : si un IF n'est pas satisfait, les clauses suivantes ne sont pas lues par BASIC
IF N>5 THEN P=10:A=B*2
Ici avec N=3 le A=B*2 n'est pas exécuté, après un IF non vérifié BASIC passe immédiatement à la ligne suivante ! Sauf s'il y a un ELSE.
IF N>5 THEN P=10 ELSE P=20
Un ELSE est très catégorique, passé cette ligne, on aura P=10 ou bien P=20. Pas d'autre alternative possible ! On peut mettre un IF après un ELSE, mais alors attention à votre raisonnement.

IF N>5 THEN P=10 ELSE IF N<3 THEN P=30

Cela se traduit par la "table de verité" suivante

N	P
1	30
2	30
3	?
4	?
5	?
6	10
7	10

A méditer

Une condition non vérifiée prend la valeur zéro, vérifiee elle vaut 1 et non pas 1. Essayez PRINT (12=2*6), reponse 1 Remplacez 12 par 13, réponse zéro Supprimez les parentheses ça marche encore.

C=(12=2*6)

IF C THEN GOSUB 3000

Ce IF C signifie "si C non nul" Vous pouvez le remplacer par IF C=-1, mais pas du tout par IF C=11

Dans le cas contraire, vous avez le choix entre

IF C=0 THEN GOSUB 4000 ou IF NOT C THEN GOSUB 4000 IF NOT C signifie "si C n existe pas s il est nul"

Pour combiner dans une même ligne des AND et des OR, il ne faut pas hésiter à utiliser des parenthèses Même si elles ne servent à rien, c est plus sûr et le listing est plus clair à lire

IF (MARQ$="AMSTRAD") AND (TYP=464 OR TYP=664 OR TYP=6128) THEN PRINT "c'est le meilleur BASIC

Un "test" par =, >, <, est donc note par 1 (vrai) ou par zéro (faux) Ceux qui aiment les maths s apercevront alors que AND est l'equivalent de * et OR l'equivalent de +

Avec de l aspirine à portée de main, vous pouvez imbriquer plusieurs niveaux de parenthèses avec des OR AND, NOT, mais alors prenez la precaution de tester votre ligne logique par un petit programme d'essai (commencant par des INPUT) et ce afin de verifier que la "table de vérité" obtenue correspond bien à vos desiderata

Attention à la fonction XOR ! Super restrictive (A=5 XOR B=1) n'est vraie (1) que si l'un seulement de ces deux tests est vrai Faux si ces deux égalites sont verifiées (ou aucune des deux)

Les sorties de boucles

Supposons une boucle FOR NEXT qui débute en ligne 300 par FOR N=1 TO 15 et qui se termine par 460 NEXT En ligne 350, un IF permet de sauter la suite du "traitement" et de passer au N suivant La syntaxe normale est

350 IF THEN GOTO 460

alors la grosse bourde (helas classique) serait d écrire

350 IF THEN NEXT

En effet, quand BASIC rencontre un FOR, il repere d abord ou est son NEXT, ici en 460 Il ne verra pas celui de la ligne 350 "masqué derrière" un IF Alors si le IF le conduit à ce NEXT en 350, il risque fort d'y avoir du vilain

Si, en ligne 380, vous prévoyez un cas permettant de mettre fin a cette boucle, donc avant que N atteigne 15 la pire des choses serait d y mettre un GOTO 470 En ce cas, il suffit d'ecrire

380 IF... THEN N=15 GOTO 460

et l on sort PROPREMENT, sans risque de choc en retour

Idem pour une boucle WHILE WEND On met la variable à la valeur attendue, et GOTO le numero de ligne du WEND

Idem pour les sous-programmes pour une sortie prématurée ne pas faire IF... THEN RETURN mais THEN GOTO le numero de l'UNIQUE RETURN

Si dans une boucle ou dans un sous programme, vous desirez faire un detour vers l'exterieur, il est formellement interdit d operer par GOTO ! Uniquement par GOSUB Sinon c'est l'affolement des "pointeurs"

On ne peut pas imbriquer des boucles WHILE WEND (dans le BASIC des CPC)

LES ERREURS OU PASSE

Ces erreurs ont deux causes :

— Présence de lignes obsoletes que l'on a oublie d'effacer, par étourderie Soit des lignes dont on a changé le numero pour les placer ailleurs soit des lignes de tests que vous aviez insérées ça et la, du genre

629 PRINT N ,A$ STOP

(utilisez toujours des numéros de lignes se terminant par 9 plus faciles à reperer !

— On s est trompe de version ! La première chose à faire, avant de modifier un programme est d'en changer la date dans le REM de la ligne 10 Et comme vous avez pris la précaution (?) d'enregistrer votre logiciel vedette de 13 kilo octets sur quatre disquettes différentes, votre derniere modification porte en fait sur l'avant-avant avant dernière version Alors .

LES LOCALISATIONS DIFFICILES

Votre programme ne plante pas, mais conduit a des resultats farfelus Donc le bug peut être n'importe ou ! Le premier travail consiste alors à localiser la zone du listing qui est coupable Par exemple ce passage de 20 lignes que l'on réduira ensuite a une Il faut jouer au detective

— Inserez des lignes mouchard comme

629 PRINT N,A$ STOP

et vous redémarrez en tapant CONT

— Testez en mode direct des sous programmes Par exemple

A=4 X=0 A$="Paul GOSUB 3000

Et là toujours en mode direct, interrogez par PRINT certaines variables

Autre variante des "données fixées"

NB=5,B$="JEAN" GOTO 540

et vous recupérez les résultats plus bas dans une ligne-mouchard Remarquez qu il ne faut pas utiliser RUN qui annule toutes les variables en cours, mais GOTO ou GOSUB qui les conserve

Deuxième degré Vos mouchards vous donnent partout des indications normales, donc le bug n'apparaît que pour certaines valeurs de la variable M Vous ne voyez pas pourquoi, mais le fait de connaître les valeurs coupables va vous mettre sur la voie

```
1 FOR M=-10 TO +20 GOSUB 5000
2 IF LEN(B$)>5 THEN STOP
3 NEXT END
```

Et on lance par GOTO 1

Troisieme degré ça y est ! Vous avez identifie ce qui provoque l'erreur, le groupe de lignes ou le mal se développe mais vous ne comprenez toujours pas pourquoi En effet, il s'agit d'un passage très complexe avec des IF ... THEN GOTO (ou GOSUB) ELSE GOTO etc Par où passe le programme ? Les armes sont TRON et TROFF (abréviations de TRACE ON et TRACE OFF)

TRON va faire inscrire à l ecran les numeros de lignes ou passe le programme (entre crochets) TROFF annule TRON Mettez très peu de lignes entre TRON et TROFF car un écran est vite rempli !

Le final

Ça y est le bug est mort, ca marche Surtout ne sauvegardez pas ! Votre programme en RAM est un champ de bataille Notez sur papier la fameuse ligne corrigée, puis NEW ! LOAD , correction de la ligne et enfin SAVE Vous voyez pourquoi ?

CONCLUSION

Le bug, c est le sel de la programmation Oue ce serait triste si tous les programmes fonctionnaient toujours du premier coup ! Oue c'est bon un bug farceur qui joue à cache-cache pendant des heures, que l'on traque en se prenant pour Columbo ... Et puis çà dispense les programmeurs de s'acheter des logiciels d aventures à 270 francs la disquette ...

BRANCHER LE TURBO

AU COEUR DU PROBLEME : LES INSTRUCTIONS

Yves GERAULT

(S U I T E)

Illustrations Jean-Luc AULNETTE

Les lignes PASCAL qui appartiennent au bloc instruction sont elles-mêmes appelées instructions. Une instruction PASCAL sera donc constituée par les caractères compris entre deux points-virgules. Comme nous l'avons vu précédemment, cette instruction peut être écrite sur une ou plusieurs lignes écran. Voyons maintenant les instructions que le concepteur de PASCAL TURBO met à notre disposition.

INSTRUCTION COMMENTAIRE

Pascal est un langage compilé Nous avons vu que le code exécutable (en binaire) est obtenu à partir du code source (en ASCII) via le passage par un traducteur (la compilation) Un point très important du fonctionnement de ce compilateur est qu'il ne s'occupe pas de ce qu'il va reconnaître comme des commentaires Nous pouvons donc en mettre autant que nous voulons dans notre texte source cela n allongera en rien notre code objet. Alors ne nous en privons pas

Le compilateur reconnaît comme commentaire tout ce qui se trouve entre des accolades Le commentaire commence avec une accolade ouvrante et se termine avec une accolade fermante Bien sûr, ce sera une erreur fréquente d'oublier cette accolade fermante, ce qui fait que le compilateur prendra tout le reste de votre programme comme un commentaire Mais cette erreur est facile à détecter avec un peu d'entraînement

Pour assurer la compatibilité de PASCAL TURBO sur des micro ne possédant pas les accolades, le concepteur autorise une autre manière de signaler les commentaires il s'agit des caractères parenthèse ouvrante suivie immédiatement par l'astérisque < < (* > > pour le début de commentaire et bien sûr de astérisque suivi immédiatement de parenthèse fermante pour la fin de commentaire.

Où peut on placer des commentaires ? A n'importe quel endroit, dans n'importe quel bloc ou sous-bloc en remplacement d'un séparateur En d'autres termes, le seul endroit ou il est interdit

de mettre un commentaire, c'est à l'intérieur d'un identificateur car, par définition, un identificateur est une suite de caracteres ne comportant pas de caractères séparateurs. Nous utiliserons largement ces possibilites en commentant très abondamment nos listings

LES OPERATEURS ARITHMETIQUES

Il est possible de faire subir à des variables, un certain nombre de traitements à l'aide de symboles appelés opérateurs arithmétiques, ils sont au nombre de 12

+ addition
 soustraction
* multiplication.
/ division
not négation bit à bit
or ou logique bit à bit.
xor xor logique bit à bit
and and logique bit à bit
shr decalage de bits à droite (shift right)
shl décalage de bits à gauche (shift left)
mod reste de la division
div division entière.

Les opérateurs + - * / peuvent agir sur des objets de type entiers ou reels, mais, attention, le résultat n'est pas toujours du meme type que les operandes (ex 3/5 = >0 B, les deux opérandes sont entiers, mais le résultat est réel). Le langage PASCAL vous signalera une erreur de fonctionnement si vous ne prevoyez pas ce genre de situation
Par contre, les operateurs arithmétiques not, or, xor, and, shr, shl, mod, div, ne peuvent travailler que sur des entiers et le résultat est un entier.
Une suite d'actions sur une suite d'objets est appelée une expression. Dans cette expression comme en BASIC, il y a un ordre de préséance entre ces operateurs
— not
— * / div mod and shl shr
— + or xor
En cas d'égalité de priorité, le calcul de l'expression se fera de gauche à droite et l'ordinateur prendra les opérateurs quand il les trouvera
Comme en basic, pour éviter les erreurs, mettez vos expressions entre parenthèses pour forcer la priorité des opérateurs.

QUELQUES IDENTIFICATEURS PARTICULIERS

Certains identificateurs ont une valeur par défaut

pi · 3 1415926536 E+00
maxint 32767 (c'est le plus grand entier possible)
true signifie vrai pour un booleen
false signifie false pour un booleen

INSTRUCTION AFFECTATION

Le résultat d'une expression est un objet qui a un certain type suivant la nature des opérateurs que nous avons fait agir. Il faut bien sûr stocker la valeur de cet objet à l'intérieur d'une case mémoire identifiée par un identificateur précisément. Ce qui revient à affecter à un identificateur le résultat de l'expression

Prenons un exemple ; si nous voulons calculer la somme de A et de B et mettre le résultat dans C nous pouvons schématiser l'opération par
AFFECTER à C le RESULTAT DE A + B
Ce qui va s'écrire en BASIC LET C = A + B ou trés malheureusement par abus de langage C = A + B Ce qui n'a aucun sens strictement parlant. Bien sûr, tous ceux qui font de l'informatique le savent, mais il faut admettre que l'écriture de I = I + 1 a de quoi choquer bien des débutants
PASCAL fait partie des langages qui n'acceptent pas ce fâcheux abus de langage. C'est pourquoi il demande à ce qu'on note cette AFFECTATION par la syntaxe C . = A + B Ce << = >> est le symbolisme réservé à l'AFFECTATION, on réservera le symbolisme << = >> à l'égalité comme en arithmétique

INSTRUCTION WRITE ET WRITELN

Tôt ou tard il faudra bien qu'un résultat sorte de notre ordinateur. Quelqu'en soit la destination, l'instruction aura la même syntaxe et elle ne surprendra pas les utilisateurs du BASIC LOCOMOTIVE puisqu'elle est très voisine. Le listing 1 en donne quelques exemples
Suivant notre habitude décortiquons le premier exemple

write mot reservé du langage PASCAL TURBO signalant que nous allons effectuer une sortie sur un périphérique
(début des attributs de ma sortie ,
output ; précise que le canal que je veux utiliser est l'écran
, separateur d'attribut
var 1 identificateur repérant la variable dont je veux afficher le résultat
, séparateur d'identificateur
var 2 idem var 1
) fin des attributs
; fin de l'instruction

Passons au deuxième exemple en ne parlant que des choses qui changent

lst précise que le canal que je veux utiliser est l'imprimante
a+b c . expression arithmetique dont le résultat sera imprimé
a shr 2 · idem

Troisième exemple

fich 1 précise que le canal de sortie sera un canal privé de l'utilisateur (un fichier disque peut-être)
123 PASCAL TURBO autorise les sorties directes de valeur.
'mot idem, mais avec une chaîne de caractéres, remarquez qu'en PASCAL, les chaînes sont encadrées par des apostrophes simples et non par des quotes " " comme en BASIC

PASCAL présente une particularité du point de vue de l'affichage. Contrairement à BASIC il n'a pas de tabulation par défaut. Il écrit ce qu'on lui demande d'écrire sur l'écran à la position ou se trouvait le curseur à ce moment. De plus, il ne laisse aucun espace entre les valeurs. C'est au programmeur de prévoir les espaces nécessaires. Cela correspondrait à l'option

<<,>> du PRINT BASIC
Certaines facilités sont cependant offertes qui permettent de solutionner tous les problèmes de mise en page.
— Si on remplace WRITE par WRITELN, l'ordinateur, après avoir écrit ce qu'on lui a demandé, positionnera le curseur sur la première colonne de la ligne suivante : il fera un retour à la ligne comme une machine à écrire.
— Prenons maintenant l'exemple numéro 4.
var 1 5 2 : signifie que le contenu de la variable var 1 sera affiché sur 5 colonnes, que le résultat sera cadré à droite avec 2 chiffres après le point décimal (il est évident que var 1 est du type real).
var 2 4 : signifie que var 2 est du type entier et qu'il sera affiché sur 4 colonnes cadré à droite.
mot 12 : en supposant que mot soit du type string, le contenu de mot serait affiché sur l'écran sur 12 colonnes.

INSTRUCTIONS READ ET READLN

Nous n'étudierons que le cas de READLN. Nous reviendrons plus tard sur l'utilisation de READ qui suppose une digression préalable sur la gestion du clavier sous PASCAL TURBO ainsi que sur la notion généralisée des fichiers (pas de quoi s'affoler, mais chaque chose en son temps).
L'ordre READLN ressemble beaucoup à l'ordre INPUT du BASIC, mais en diffère aussi notablement. Des exemples sont donnés par le listing 2.
La différence essentielle est qu'en PASCAL un ordre READ, cela ne sert uniquement qu'à lire et qu'un ordre WRITE cela ne sert uniquement qu'à écrire. C'est assez logique, mais ce n'est pas le cas en BASIC. La signification des divers champs de cette instruction est assez évidente, rappelons les rapidement.

readln : mot réservé du PASCAL TURBO signalant au compilateur qu'une entrée va être faite.
(: début des attributs de l'instruction.
input : désigne que cette entrée va être faite par le canal entrée standard (le clavier).
, : séparateur d'attribut.
var 1 : identificateur de la variable à laquelle va être affectée la valeur de ce qu'on rentrera.
var 2 : idem var 1.
) : fin des attributs.
; : fin de l'instruction.

Autre différence notable, les diverses valeurs devront être séparées par des espaces et non des virgules comme en BASIC. Cette différence est plus gênante car, en cas d'erreur en entrée, il n'y a qu'une sanction : la pire !! <<run time error>> et arrêt de l'exécution du programme. Il y a bien sûr, moyen d'éviter ce plantage du programme, mais ce n'est pas encore à notre niveau. Alors, faites très attention à la syntaxe de vos entrées. Bien sûr, puisque cette instruction se finit par un <<ln>>, le curseur se déplace au début de la ligne suivante après avoir terminé sa lecture.

FONCTIONS ET PROCEDURES

En dehors des opérateurs qui nous permettent déjà de calculer diverses expressions et d'en affecter le résultat dans une variable repérée par son identificateur, il est possible d'effectuer d'autres types d'actions qui se nomment des fonctions et des procédures.
LES FONCTIONS sont des actions déjà connues de ceux qui ont travaillé en BASIC. Certaines portent même le même identificateur par exemple <<cos>> pour cosinus, d'autres en diffèrent par exemple <<length>> au lieu de <<len>> pour longueur d'un mot. Toutes ces actions ont un point commun : elles ne délivrent qu'une seule valeur qui doit être affectée à une variable (ex : cosinus 20 degré = cos (20)).

LES PROCEDURES ne sont pas totalement inconnues à l'utilisateur du BASIC. Prenons comme exemple <<locate 20,10>>. En PASCAL TURBO, elle sont généralisées et plus performantes. Citons pour exemple <<str(valeur mot)>> qui convertit une <<valeur>> numérique entière en un <<mot>>, une autre <<gotoxy (25,30)>> aura le même sens que notre <<locate>> en BASIC.
Ces fonctions et procédures sont nombreuses et nous nous attarderons dans notre prochain article à vous les décrire. Mais où le PASCAL devient génial, c'est que s'il n'y en a aucune qui vous plaise pour résoudre votre problème, PASCAL vous autorise à la créer à votre guise et ensuite à vous en servir comme si elle avait toujours existé (pratique non!!!). Nous y reviendrons très prochainement.
Pour clore cet article sur les entrées sorties simples, le listing 3 vous montre ce qu'on peut faire avec les quelques instructions que vous avez maintenant à votre disposition.

Listing 1 : Exemple d'instruction write et writeln

```
{**************************************************************}
{ ceci est un commentaire puisque compris entre des accolades }
{**************************************************************}

{ exemple numero 1}
write (output,var_1,var_2 );

{ exemple numero 2}
write (lst,wth-c,a s r 2 );

{ exemple numero 3}
write (lcr 1,1- , e t );

{ exemple numero 4}
writeln (output,var_1:5:2, a  4 , 1 );

(**************************************************************)
(* ceci est aussi un autre commentaire puisque compris entre *)
(*      les symboles ( suivi immediatement par *             *)
(*                 et * suivi immediatement par )            *)
(**************************************************************)
```

listing 2 Exemple d'instruction readln

```
=================================================================
!                                                               !
!    ( exemple numero 1 )                                       !
!    readln (input,var_1,var_2);                                !
!                                                               !
=================================================================
```

listing 3 Programme Pascal illustrant les notions développées dans cet article

```
=========================================================================
program essai_des_operateurs_arithmetiques (input,output);
{=====================================================}

{ debut de mon bloc declaration}
  { debut de mon sous bloc declaration des constantes }
   const
         constante_entiere   = 100;
         constante_reelle_1  = 1.25E+05;
         constante_reelle_2  = 12.354;
         constante_string    =  mot constant ;
  { fin de mon sous bloc declaration des constantes }

  { debut de mon sous bloc declaration des variables }
   var
         variable_entiere_1,
         variable_entiere_2    :   integer ;
         variable_reelle_1,
         variable_reelle_2     :   real ;
         variable_string       :   string (80);
  { fin de mon sous bloc declaration des variables }
{ fin de mon bloc declaration }

{ debut de mon bloc instruction }
 begin
   write (output, entrer valeur des variables entieres 1 et 2 : );
   readln (input,variable_entiere_1,variable_entiere_2);
   write (output, entrer valeur de variable string : );
   readln (input,variable_string);
   variable_reelle_1:=2 * constante_reelle_2;
   writeln (lst);writeln (lst,     . , debut de l execution );writeln(lst);
   writeln (lst,     . ,variable_reelle_1,constante_reelle_1);
   writeln (lst,     . ,variable_reelle_1:9:5,constante_reelle_1:10:2);
   writeln (lst,     . ,variable_entiere_1:5,variable_entiere_2:5);
   writeln (lst,     . , maxint =  ,maxint:10, pi =  ,pi:20:15);
   variable_entiere_1:=9;variable_entiere_2:=2;
   writeln (lst,     . , 9 mod 2 = ',variable_entiere_1 mod variable_entiere_2:3);
   writeln (lst,     . , 9 div 2 =  ,variable_entiere_1 div variable_entiere_2:3);
   writeln (lst,     . ,0000 0001 or 0000 0010 or 0000 0011 = 0000 0011 );
   writeln (lst,     . ,    1     or     2     or     3     =     3     );
   writeln (lst,     . ,                                             ,1 or 2 or 3);
   writeln (lst,       ,constante_string:40, *** fin de la zone );
   variable_string:=constante_string+  plus long ;
   writeln (lst,     . ,variable_string);
 end.
{ fin de mon bloc instruction et fin de mon programme}
```

```
.debut de l execution

.  2.47080000000000E+001  1.25000000000000E+005
. 24.70800 125000.00
.    9    2
.maxint =       32767pi =    3.14159265358979O
. 9 mod 2 =   1
. 9 div 2 =   4
.0000 0001 or 0000 0010 or 0000 0011 = 0000 0011
.    1     or     2     or     3     =     3
                                          3
.                               mot constant*** fin de la zone
.mot constant plus long
```





PHOTO-SPRITE



PHOTO-SPRITE est une "boîte à outils" permettant la gestion de mini-images graphiques de 32*32 points (genre photo d'identité).

Elle vous permettra de créer et de reproduire des photos, blasons ou représentations d'objets pour agrémenter un jeu d'aventure.

L'ensemble comprend :

— une routine d'affichage en langage machine baptisée PLDT1 ou PLOT2 suivant son adresse d'implantation, qui sera générée par le programme BASIC CHARGEUR (listing n° 1) ;

— deux utilitaires CONVERT et GENEPHOT pour générer les fichiers PHOTOx.BIN contenant les images à visualiser.

Le programme fonctionne sur CPC 464, 664 ou 6128.

LA ROUTINE PLOT1

Ecrite en langage ASSEMBLEUR pour avoir un affichage très rapide, elle est appelée à partir du BASIC par un CALL &A100 après avoir fixé l'origine de la fenêtre graphique. La routine peut reproduire des images en mode 1 avec quatre couleurs et en mode 2 avec deux couleurs.

Elle occupe les adresses &A100 à &A170, l'image occupe 1024 octets de &A200 à &A5FF, l'espace restant entre les deux est utilisé par la routine pour ses variables.

Si vous désirez faire défiler plusieurs photos au même emplacement, vous pouvez appeler la routine par CALL&A133, l'initialisation de la fenêtre n'étant pas à refaire.

Dans le cas où vous auriez des extensions connectées autres que le lecteur de disquette (synthétiseur, interface RS232C...) la routine sera en conflit avec l'espace mémoire réservé pour ces dernières. Les connaisseurs pourront facilement reloger la routine en modifiant les adresses de saut ; pour les autres, le chargeur BASIC de la routine vous proposera deux adresses d'implantation &A100 ou &8100. Si vous avez des extensions choisissez &8100 mais pensez à modifier en conséquence les programmes BASIC.

LE PROGRAMME CONVERT : (listing 2)

Vous permet de créer un fichier PHOTOx.BIN à partir de données en DATA. Conservez les DATA de vos créations, si vous effaciez votre fichier .BIN c'est le moyen le plus rapide de le reconstituer. Seules les photos en 2 couleurs peuvent être traduites en DATA (ces 2 couleurs correspondant aux ink 0 et 1 de préférence), lors de la traduction, toute couleur différente de l'ink 0 est considérée égale à 1.

LE PROGRAMME GENEPHOT : (listing 3)

L'outil indispensable pour donner libre cours à votre talent.

— la saisie s'effectue directement à l'écran, le curseur est déplacé par les flèches ou avec le joystick. La case pointée prend la couleur du crayon sélectionné en pressant la touche COPY ou le bouton FEU du joystick.

Les commandes disponibles sont affichées dans une fenêtre :

— "Load" charge une photo après vous en avoir demandé le numéro. La photo sera affichée à l'échelle 1 et reproduite dans la grille de saisie pour être éventuellement modifiée ;

— "Save" sauvegarde une photo. Lors de l'appel de cette fonction, une mise à jour de l'image à l'échelle 1 est effectuée automatiquement. Un sous-menu vous permet la sauvegarde, l'édition des DATA (pour photos en 2 couleurs uniquement) ou le retour au menu principal ;

— "Couleurs" appelle un sous-menu qui vous permet d'adapter l'écran à votre goût et de choisir les couleurs des crayons. Les numéros de couleurs sont affichés en regard des commandes de sélection pour le bord et les quatre encres ;

— "Aller case" déplace le curseur à l'emplacement demandé ;

— "Mise à jour" code en mémoire la photo de la grille et affiche le représentation à l'échelle 1 correspondante ;

— 0,1,2,3 sélectionnent le crayon actif pour la saisie, ce dernier est repéré en permanence par une flèche ;

— pour les 2 dernières commandes, "Effacer" et "Quitter", une confirmation est demandée.

NB : un BREAK peut se produire si vous demandez le chargement d'une photo ne figurant pas sur la disquette, ce type d'erreur n'étant pas interceptable en BASIC, il vous faudra relancer le programme par RUN 520.

ADAPTATIONS DE LA ROUTINE PLOT

La routine peut reproduire également les photos en mode 2 (seulement celles générées en 2 couleurs).

L'affichage en 2 couleurs peut être accéléré en pokant des 0 en &A14B, &A14C, &A140.

Vous pouvez changer les dimensions de la fenêtre graphique, mais il vous faudra également modifier GENEPHOT ainsi que CONVERT et tenir compte de la taille mémoire des fichiers PHOTOx.BIN.

En conclusion, cet ensemble devrait vous permettre de donner à vos programmes un "look pro" et vous inciter à vous initier à l'Assembleur, dans ce dernier cas deux ouvrages vous seront nécessaires :

— le FIRMWARE de votre CPC qui vous donnera toutes les informations sur les routines utilisables de la ROM ;

— PROGRAMMATION DU Z80 de RODNAY ZAKS aux Editions SYBEX qui est la véritable bible pour ce microprocesseur.

LOGICIELS



LES PASSAGERS DU VENT 2
INFOGRAMES
Aventure

Durant une période de trois mois, vous avez pu tomber sous le charme des Passagers du Vent. Maintenant, il est temps de passer à la suite qui d'ailleurs n'est certainement pas la fin.
Dans ce deuxième épisode, nous retrouvons Isa, héroïne principale, qui est une jeune libertine entreprenante en cette fin de XVIIIᵉ siècle. Elle a entrepris de voyager dans le monde entier (des ports bretons aux côtes africaines) afin de récupérer un titre qui lui a été usurpé lorsqu'elle était petite : celui de comtesse.
Avec "l'heure du serpent", nous nous retrouvons dans le village du roi KPENGLA. Isa va devoir affronter aussi bien les mystères vaudous que les fauves de la savane...
Nous retrouvons avec joie un graphisme toujours aussi éclatant et envoûtant avec, comme dans le premier épisode, une musique superbe changeant avec chaque scène... Cette fois, il y a huit scènes différentes avec un petit plus : en effet, à la fin des cinq premiers tableaux, le sage vous pose une question sous forme d'énigme ; à vous de sélectionner le bon personnage pour donner la réponse et obtenir ainsi un indice pour l'issue finale de l'histoire.
Dernière nouveauté par rapport au numéro 1 : en plus du défilement des légendes obtenues en sélectionnant les icônes des personnages, vous pouvez intervenir parfois sur la partie graphique et utiliser ainsi certains objets.
En conclusion, la série des Passagers du Vent doit avoir sa place dans votre "logithèque".



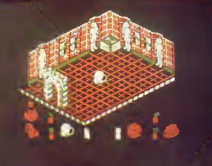
LE CHEVALIER BLANC
COBRA SOFT
Arcade/Aventure

Avec cette nouvelle production de Cobra, vous entrez dans une histoire de type moyenâgeux avec le roi, la princesse et le beau chevalier servant qui, de surcroît, est blanc pour l'occasion.
Tout commence lorsque la princesse et le Chevalier Blanc se promènent tranquillement. Ils se font attaquer par d'étranges créatures qui, non contentes d'enlever la princesse, capturent également les armes du Chevalier.

Celui-ci n'a alors d'autre but que d'aller délivrer la belle dans un étrange château après avoir récupéré ses armes. Pour cela il doit commencer par capturer un cheval blanc... Ensuite, il devra traverser les marais en veillant bien à ne pas se trouver sur le chemin d'une étrange créature blanche car, si tel était le cas, il se retrouverait à son point de départ. Pouvant récupérer ses armes dans les marais, il ne lui reste plus qu'à pénétrer dans le château et à terminer l'histoire en héros ! .
Ce jeu requiert une certaine habileté au joystick. Si vous possédez cette qualité, vous pouvez passer un agréable moment avec ce logiciel "mignon" tant au niveau du scénario qu'au niveau du graphisme ou de la musique.

BALLBLAZER
ACTIVISION
Arcade/Simulation

Encore un logiciel présentant un jeu se passant dans le futur, mais possédant deux éléments traversant tous les temps : un ballon et des poteaux de buts !...
Le championnat Interstellaire de Ballblazer se passe dans les années 3000... et quelques poussières. Le but de la finale est de devenir le roi de tous les jeux. Pour y parvenir, vous vous retrouvez sur le terrain face à votre adversaire. L'écran de votre Amstrad vous permet d'avoir chacun votre propre vision du jeu puisqu'il est divisé en deux parties. Vous disposez de 3 mn (ou moins si vous le désirez !) pour prouver que vous êtes le meilleur. Après avoir capturé la balle, il ne vous reste plus qu'à vous précipiter au fond du terrain pour marquer un but et de renouveler l'opération dix fois de suite ! Alors peut être aurez vous mérité le titre de "Masterblazer".
Avec Ballblazer, vous avez entre les mains la possibilité d'être un as de la balle grâce à des regles de jeu simples et à une bonne animation. Par contre, on ne peut pas dire qu'il y ait eu une très grande recherche graphique.



HEAD OVER HEELS
OCEAN
Arcade/Aventure

Avec ce logiciel, vous devrez vous préparer à entrer dans un système solaire. L'Empire de Dentnoire est constitué de quatre planètes Egyptus, Pénitentier, Safari et le Monde du Livre. L'Empereur y fait régner un régime plus qu'oppressif mais personne n'ose se rebeller et c'est bien ce qui inquiète une planète voisine nommée bien sûr Liberté.
Les habitants de Liberté sont composés de deux parties : la tête et les jambes et elles peuvent faire des actions en étant unies ou séparées. Votre mission consiste à réunir les deux parties d'un habitant de Liberté qui sont enfermées dans le château de Dentnoire. Une fois cette première mission réalisée, il vous sera peut être possible de provoquer un soulèvement et de mettre fin à ce régime diabolique !
Avec Head over Heels, vous évoluez dans un nombre très élevé de pièces avec des pièges à profusion. Vous avancez dans un beau graphisme en 3D, très riche en détails et avec des couleurs superbes, une garantie pour passer un bon moment... d'autant plus que le jeu risque fort de ne pas être terminé en 1 heure !

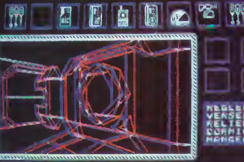
KRAKOUT
Gremlin Graphics
Arcade

En regardant le premier écran, vous vous dites : encore un casse briques ! Ne vous fiez pas aux apparences, car elles sont bien souvent trompeuses. C'est entendu ! Il y a bien une balle et des briques, mais ce jeu est tout simplement diabolique.
Ce qui frappe en tout premier lieu, c'est la multitude de couleurs qu'offre chaque écran. De plus, le dessin formé par les briques change à chaque niveau. Mais ce n'est pas tout ! En effet, lorsque vous touchez certaines briques, elles subissent un phénomène de retournement et laissant alors apparaître des lettres. Il y a neuf possibilités différentes, mais vous avez, par exemple, le E qui fait agrandir la batte ou le X qui vous apporte une vie supplémentaire. Vous me direz que tout ceci est déjà fabuleux. Mais Krakout fait encore mieux car vous avez des options. En effet, au départ, la batte se trouve à droite, mais vous pouvez essayer de jouer avec la batte à gauche (pas si évident que cela !). Enfin, derniers points importants : vous pouvez agir sur la vitesse de la balle, mais également sur celle du déplacement de la batte. De cette manière, chacun peut se régaler suivant ses propres capacités, d'une part, et s'améliorer pour devenir un crack, d'autre part.

RELIEF ACTION
LORICIELS
Aventure

Ce nouveau logiciel veut nous apporter une nouvelle dimension au jeu d'aventure. Quelle est elle ? Pensez-vous... Il s'agit tout simplement de relief dynamique ! Campons tout d'abord le décor : vous faisiez partie d'une équipe scientifique en mission sur la planète VEGA. Malheureusement, durant le vol, une mauvaise manipulation du laborantin a créé un monstre... Et, désormais, vous êtes le seul rescapé de l'équipe. Vous allez devoir déambuler dans tout le vaisseau (rassurez-vous, il n'y a que trois niveaux et une cinquantaine de pièces !), afin de rassembler pièces et indices pour atteindre la navette. Bien entendu, il est un objet que vous devez absolument vous procurer pour que le monde entier ne doute pas de vos propos : il s'agit du carnet de bord du commandant...
Relief Action nous propose des écrans avec le dessin des pièces sous forme filaire. De plus, la vue en 3D peut être simplifiée en sélectionnant l'option relief qui fait apparaître des traits dédoublés bleus et rouges. Dans ce cas, vous devez utiliser la paire de lunettes qui est livrée avec le logiciel. Ceci





est un aspect intéressant du jeu, mais il faut également considérer que vous suivez un déplacement du décor en temps réel, ce qui vous permet de vraiment participer à l'aventure Somme toute, un produit très intéressant à étudier, avec ou sans relief .

SIGMA 7
UBI SOFT
Arcade

Comme son nom l'indique, ce jeu se divise en 7 étapes, chacune d'elles comportant trois phases



Vous êtes aux commandes d'un vaisseau et vous devez faire face, avec réflexes et vivacité, à tout ce qui se présente devant vous, sans aucune exception, car vous n'avez vraiment pas le temps de vous poser la question : ami ou ennemi ?

Vous devez tout d'abord atteindre une usine en vous frayant un chemin entre les mines de défense aérienne (la distance à parcourir grandit à chaque étape...) Une fois arrivé "à bon port", vous déblayez des sentiers sur lesquels des robots mines se précipitent sur vous de façon inamicale. Avant de sortir de cette phase, retenez bien le motif que présentent certaines pastilles car il vous sera fort utile lors de la phase finale... Dans celle-ci, vous devrez être très rapide pour reconstituer le dessin , cette partie est vraiment très délicate

Avec ce logiciel, vous trouverez une bonne séance d'entraînement aux réflexes avec, de surcroît, un graphisme et un son agréables...

SCALEXTRIC
FIL
Simulation

A cette époque où la saison des Grands Prix a recommencé, vous pouvez tester votre habileté à tenir un volant de Formule 1 avec ce logiciel.

Vous avez la possibilité de faire la course contre l'ordinateur ou contre un adversaire en chair et en os (malheur à lui s'il hérite des commandes au clavier !).

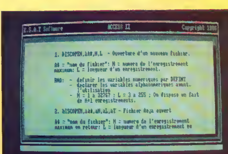
Pour commencer, il faut sélectionner les conditions dans lesquelles vous voulez concourir : après le choix du circuit (vous avez 17 circuits de Formule 1 à votre disposition), il ne vous reste plus qu'à indiquer le nombre de tours que vous désirez couvrir et la force de votre adversaire

De plus, si aucun circuit ne vous satisfait, vous avez la possibilité de construire votre propre parcours et de le sauvegarder sur cassette ou disquette... Ainsi donc, les possibilités de courses sont pratiquement illimitées.

L'écran se divise en deux parties, permettant ainsi de visualiser la course de chaque concurrent ; de plus, la vue totale du circuit montre clairement les positions respectives des concurrents... Avec Scalextric, vous avez entre les mains une bonne simulation, avec un graphisme plus que correct







ACCESS II
E S A T SOFTWARE
Utilitaire

Le lecteur de disquettes intégré à l'Amstrad (6128, 664) ou bien séparé de l'unité centrale (464) est bien utile à ceux qui veulent traiter des fichiers. Malheureusement, les CPC ne sont pas pourvus d'une fonction fort intéressante l'accès direct. Imaginons que vous ayez un fichier de 30 Ko, la recherche d'une fiche se fait par le chargement de tout le fichier en mémoire. Avec l'accès direct, la

recherche se fait sur le disque et seule la fiche concernée est mise en RAM. Cela entraîne un gain de place conséquent et limite la taille des fichiers à la capacité d'une face de disquette (170 Ko environ).

Le programme Access II vous propose une gestion de fichiers en accès direct par l'intermédiaire de routines RSX. Les nouvelles instructions sont au nombre de 5 :

IDISCOPEN : son action est identique à celle de "OPEN" en BASIC Amstrad. Elle possède 3 paramètres : le nom du fichier, le nombre d'enregistrements maximum et la longueur des enregistrements (de 3 à 255). En effet, avec l'accès direct, toutes les fiches doivent avoir la même taille. Cela permet de les retrouver plus facilement.

IDISCLOSE : on le devine, il s'agit de la fermeture du fichier. Celle-ci est obligatoire, sinon il y a perte de données.

IDISCWRITE : après avoir ouvert un fichier, il faut écrire les données. DISCWRITE comprend 3 paramètres : un code d'erreur R, le contenu de l'enregistrement et le numéro de l'enregistrement. Le code d'erreur signale une anomalie survenue après l'utilisation de IDISCWRITE.

IDISCREAD : il s'agit de la routine de lecture des données. Elle agit de la même façon que IDISCWRITE et possède le même nombre de paramètres.

Dernière instruction : IDISCFIND permet de faire des recherches dans un fichier. Il suffit de définir quatre variables : R le code d'erreur, le contenu de l'enregistrement, le premier numéro de la recherche et le dernier numéro. La recherche peut porter sur l'enregistrement ou sur le numéro.

Un programme de démonstration se trouve sur la disquette et vous donnera un exemple de ce qu'il est possible d'obtenir avec l'accès direct. Le menu vous propose la création d'un fichier, quelques utilitaires : chargement de USER, de lecteurs, augmentation de la vitesse du moteur. Un éditeur de secteur vous permettra de modifier des fichiers en altérant directement le contenu du disque.

2 exemples, lecture et recherche sur un fichier, vous permettront d'apprécier les capacités du programme. Transferts concerne les échanges de fichiers entre lecteurs. Aide et Quitter ne nécessitent pas d'explications quant à leurs fonctions. Pour utiliser Access II sur vos disques, il faudra charger le programme DISCMAN par MEMORY &9470-1:LOAD "DISCMAN", puis initialiser en tapant CALL &9470. Je me permets de décrire ces opérations puisqu'elles ne sont pas précisées en clair dans la notice. Un logiciel simple à utiliser et qui vous ouvre de nouveaux horizons.

IMPRESSION
E S A.T SOFTWARE
Utilitaire

Lassé de corriger vos programmes à l'écran, vous êtes devenu possesseur d'une imprimante flambante neuve. La sortie de listing ou l'impression d'un message sur papier ne posent pas de problèmes grâce à LIST # 8 et PRINT # 8. En revanche, les justifications, les tabulations, les différents styles de caractères sont beaucoup moins accessibles. Il semble en effet illusoire de parvenir à emmagasiner les suites interminables de CHR$ même avec le manuel comme support.

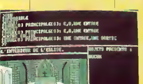
Le logiciel "IMPRESSION" vous offre une série de RSX (une soixantaine) permettant toutes sortes d'acrobaties imprimables. Le nombre de nouvelles instructions peut vous effrayer, mais le nom de chacune d'entre elles a été choisi de manière à suggérer leur fonction. Ainsi IITAL provoque l'impression en "italique" alors que IAITAL supprime ce style de caractères. Chaque fonction possède son opposé signalé par le préfixe "A". Les tabulations horizontales et verticales ont aussi leur RSX précédant en cela les avances du papier et les sauts de page.

Quelques instructions plus exotiques : IBIP 1 qui fait sonner le buzzer plusieurs fois, IVNID qui règle l'imprimante en mode unidirectionnel.

J'ai gardé le meilleur pour la fin : la recopie d'écran. COPYTEXTE, comme vous l'avez deviné, effectue une recopie rapide de l'écran texte. COPYGRAPHIQUE permet une "hard copy" de l'écran graphique en deux couleurs (encre et fond). Cette recopie est satisfaisante en MODE 2 où le nombre de couleurs est limité. Si l'écran utilise les 16 couleurs disponibles (MODE 0), il sera difficile de rendre toutes les nuances et les dégradés avec de l'encre noire. C'est pourquoi une fonction "TRAME" a été ajoutée permettant de coder les couleurs sous forme de motifs géométriques.

Sur un écran monochrome, on utilise ICOULEUR ou INTRAME pour déterminer le numéro des couleurs présentes. Il ne reste plus qu'à lancer ICOPYTRAME ou INCOPYTRAME pour obtenir sur papier un dessin "coloré". Si vous n'êtes intéressé que par une portion de l'écran, COPYPART imprimera le motif choisi auparavant. Il ne reste plus qu'à faire une impression énorme en sélectionnant DOUBLECOPY. Vous obtiendrez une image "géante" sur deux feuilles qu'il vous faudra réunir.

Impression vous facilitera sans doute la vie. En revanche, il vaut mieux déjà avoir manipulé une imprimante car les explications fournies dans la notice ne vous seront pas d'un grand secours.

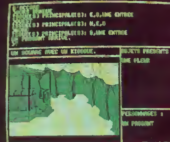




TOP SECRET

LORICIELS
Aventure

Avec cette aventure, vous êtes chargé d'une mission "Top secret" qui peut se révéler vraiment très dangereuse. Vous apprenez à la radio qu'il vient de se passer un coup d'Etat et que le Président a été enlevé.
Vous connaissez un des membres d'une organisation secrète qui a décidé de le sortir de là : il s'appelle Manu. Malheureusement, celui-ci est arrêté ! Dès lors, il ne vous reste plus qu'à déambuler dans toute la ville à la recherche d'indices et de contacts intéressants afin de récupérer le matériel nécessaire pour l'évasion, d'une part, et surtout l'indispensable mot de passe.
La version de ce logiciel sur PCW est vraiment très réussie au niveau du graphisme car l'écran monochrome n'altère en rien l'intérêt du jeu. Le problème se situe au niveau du vocabulaire à utiliser ou non car, en effet, il n'y a aucun guide en haut de l'écran comme c'était le cas dans la version pour les CPC. De toute façon, il faut employer l'impératif et vous avez, malgré tout, une aide qui vous apportera (peut être !) quelques indices lorsque vous serez par trop désespéré.



ORPHEE

LORICIELS
Aventure

La descente aux Enfers a été longue, très longue, pour parvenir sur PCW. Mais ça y est, vous êtes enfin arrivé aux fins fonds des profondeurs et il ne vous reste plus qu'à effectuer votre voyage aux Enfers.

Votre histoire a commencé tout bêtement, comme beaucoup de faits divers : vous avez eu un accident de voiture !... Vous avez alors ressenti un vide immense en vous, puis une chaleur de plus en plus intense jusqu'à devenir insupportable. Et c'est à ce moment que la rencontre s'est effectuée : Satan était là, devant vous, "en chair et en os" !...

Le face à face avec Satan est le but de ce fantastique jeu d'aventure, mais, avant d'en arriver là, il vous faudra parcourir près d'une centaine d'endroits différents dans lesquels vous n'aurez pas moins de 24 objets à prendre et qui vous seront indispensables pour arriver au bout du voyage.

La qualité du graphisme sur PCW est la même que pour Top secret, c'est-à-dire très bonne. Ce mois-ci, vous avez donc la possibilité de vivre deux grandes aventures.



THE ADVANCED OCP ART STUDIO

RAINBIRD
Utilitaire

Rainbird avait déjà frappé un grand coup avec The Music System Il récidive en publiant un utilitaire de dessin réservé aux grosses mémoires, c'est à-dire aux 6128 ou aux 464 et 664 équipés d'une extension 64 K
Après l'introduction d'un mot de passe choisi aléatoirement dans le manuel, on passe aux choses sérieuses Une barre de menu apparait au-dessus d'un écran blanc Je vous propose, pour commencer, une petite manipulation Manœuvrez le curseur grâce au joystick ou aux touches fléchées Le déplacement est parfait : pas de clignotements La juxtaposition se fait sans bavures A propos de déplacements, la souris AMX peut être gérée par le programme Avec la bebette soux les doigts, on peut presque se sentir dans la peau d'un propriétaire de Mac Pointez donc l option MISC. Cliquez avec le bouton de tir ou la barre espace. un menu déroulant vous propose, entre autres, le choix du mode écran Sélectionnez le MODE 0 Puis, allez jusqu'à l'option FILE, faites un catalogue et choisissez le fichier RAYTRACE. Le dessin qui apparaitra alors ne vous laissera plus aucun doute sur les capacités graphiques de l'Amstrad
Voyons une à une les options offertes
PRINT . c'est une recopie d'écran sophistiquée. Jugez en plutôt : l impression peut se faire en simple ou double densité avec la restitution des couleurs par tramage , on peut agir sur les justifications les marges et le centrage, la taille en X et en Y de la recopie Le "DUMP" est possible sur des fenêtres Le choix de l'imprimante est possible entre DMP 1, DMP 2000 ou EPSON
FILE : cette option autorise toutes les manipulations habituelles des fichiers : effacement, catalogue, chargement, sauvegarde Les images peuvent être composées ou non La capacité maximale d'images sur disques est de 56 fichiers et il est possible de charger 10 dessins simultanément en mémoire Il est préférable de sauver les encres de vos œuvres grâce à l'option Palette, vous récupérerez ainsi les couleurs d'origines lors d'un chargement ultérieur
PALETTE : le choix des encres est effectué dans cette partie, avec la possibilité de produire des couleurs clignotantes
PAINT : permet de sélectionner les "crayons" employés, la forme des stylos (parmi 16 possibles), la forme du jet de l'aérographe (6 options) et sa vitesse
Les pinceaux sont représentés par des motifs très colorés On trouve un œil, une bouche un mur de briques et quelques dessins plus abstraits Si l'une des 15 brosses ne vous convenait pas, il suffirait d'utiliser l'éditeur de formes pour créer un motif personnalisé.
MISC est l'abréviation de miscellaneous (divers). Effacement de l'écran, visualisation de l'écran entier différents modes graphiques, protection des encres (utile lors du changement de mode), ainsi que l'entrée de paramètres supplémentaires (type de souris, vitesse du curseur, affichage des coordonnées) sont disponibles dans ce menu J'allais oublier la sauvegarde du programme sur disque
UNDO : est la commande miracle. En cas d'erreur, elle efface la dernière action menée.
WINDOWS : c'est une des options les plus riches Elle permet de définir les encres de la fenêtre, de choisir la taille de cette fenêtre, d'exclure certaines couleurs
L'option MULTIPLE exécute des actions répétées sur les fenêtres telles que : rotations (1/4, 1/2, 3/4), échange des encres, re-dimensionnement Les options couper copier coller "charcutent" l'écran Dernières options : chargement, sauvegarde de fenêtres et modes de recouvrement (OR, XOR, AND).
FILL : cette fonction remplit des surfaces fermées ou des couleurs d'une teinte ou d'un motif choisi à l'avance Elle possède quelques variables telles que le remplissage total ou par trame Ici encore, on trouve un éditeur permettant la création des motifs
MAGNIFY : elle permet 3 grossissements : X2, X4, X8 avec déplacement sur tout l'écran et modification au pixel près, avec ou sans grille (en mode X8).
TEXT : c'est ici que l'on définit le format des lettres à inscrire sur l'écran Il est possible d'aller jusqu'à trois fois la taille normale, d'écrire dans toutes les directions, de changer une police de caractères ou de créer un nouveau jeu
SHAPES : les formes suivantes sont présentes : ligne continue, rectangles, triangles, rayons, cercles, ellipses, le tout avec option élastique ou non.
Je pourrais continuer ainsi la liste des options pendant plusieurs pages OCP est un des logiciels graphiques les plus perfectionnés que je connaisse Sa facilité d'utilisation et sa puissance vous permettront, sans nul doute, d'atteindre des sommets dans l'art pictural sur AMSTRAD.



TRUCS ET ASTUCES

POST SCRIPTUM

Daniel VANDENBROUCKE, l'auteur de PARTICIP CPC HS n° 3, nous fait part d'une petite modification à apporter à son programme pour qu'il fonctionne parfaitement.
Reécrire la ligne 480 comme suit :

480 DATA " Les chiots que j'ai (caresse) étaient mignons.", caressés,oui,1,chiots,oui,oui

Ce qui revient à dire que les "non,non" de la version précédente deviennent des "oui,oui".

TRONCATURES

Philippe LHOSTE, du CTISTAT de SATORY, fait un sort à certaines bogues. De même, explique t il, que 1/3 n'a pas de représentation limitee en décimales, certains nombres ont un nombre de décimales indéfini en binaire, d'ou troncatures et erreur. Ce défaut est inévitable et commun à toutes les machines.

FORMULES

Généreux, Philippe LHOSTE vous offre d abord une simplification des fonctions ARCSIN(x) et ARCCDS(x) (les plus courtes sont les meilleures), que nous vous avions proposees dans un précedent numéro, ensuite les formules hyperboliques

```
ARCSIN(x) = ATN(x/SQR(1-x*x))
ARCCOS(x) = pi/2 ARCSIN(x)
SINH(x) = (EXP(x)-EXP(-x))/2
COSH(x) = (EXP(x)+EXP(-x))/2
TANH(x) = (1-EXP(-2*x))/(1+EXP
          (-2*x))
ARGSINH(x) = LOG(x+SQR(x*x+1))
ARGCOSH(x) = LOG(x+SOR(x*x-1))
ARGTANH(x) = LOG((1+x)/(1-x))/2
```

Et on est prié de ne pas se mélanger les pinceaux dans les parenthéses !

FORMATEUR, EXPLOITEUR !

Non, il ne s'agit pas du nouveau slogan entendu entre Bastille et Nation, mais tout simplement de deux petites routines qui vous feront gagner de la place sur vos disquettes. De 178 Ko, vous passerez, d'un seul coup d'un seul (enfin presque), à 191 Ko, soit un gain de 13 Ko. Voyons d'abord le formateur qui récupère les pistes 40 et 41

```
10 FOR N=1 TO 51
20 READ A$:POKE X,VAL("&"+A$)
30 NEXT N
40 CALL &0F80
50 DATA 6,9,3E,C1,21,b3,F,23,23,77,
   23,36
40 DATA 2,23,3C,10,F6,AF,F5,21,b3,
   F,E5
60 DATA 11,4,0,6,9,77,19,10,FC,57,
   1E,0
60 DATA E1,E,C1,DF,b0,F,F1,3C,FE,
   2A,20
70 DATA E3,C7,42,C0,07
```

Ne soyez pas surpris si le programme fait un RESET aprés formatage (DATA) d'une face. Voyons maintenant l'exploiteur qui doit être, impérativement, exécute avant toute opération sur le disque. Laurent KUTIL, qui nous envoie ce truc, recommande de l'incorporer à vos LOADERS

```
10 OPENOUT"X"
20 POKE &A8A8,&FF
30 POKE &A895,PEEK(&A895)+13
40 CLOSEOUT
```

Pour la vérification, executez l'exploiteur et faites CAT.

INTERVALLES

31 décembre 1999 - 10 juin 1987. Oui, c'est une soustraction ! Sauvez vous, rapidement et sans trop vous casser la tête, calculer le nombre de jours qui nous séparent de l'an 2000 ? Gerard BUGEL de BRIVE, lui, entre les données sur son 6128 et obtient, instantanément, le nombre exact. De plus, vous saurez, avec le méme programme, quel jour sera le 1[er] janvier 2000. Le programme est concentré, mais bonjour les fonctions logiques ! Soufflez dans l'alcootest avant de vous mettre au clavier. Un dérapage est si vite arrivé !

```
10 'fnj% = n° du jour, fnm% = n° du
   mois
20 'fna% = n° de l'année
30 'ncj = code du jour
40 'fnjour$ = jour de la semaine
50 DEF FNj%=VAL(LEFT$(date$,2))
60 DEF FNm%=VAL(MID$(date$,
   4,2))
70 DEF FNa%=VAL(RIGHT$
   (date$,4))
80 DEF FNjour$=jour$(FNcj+(FIX
   (-FNcj/7)*7))
90 DEF FNcj=365*FNa%+FNj%+
   31*(FNm% 1)+(ABS(FNm%
   <3)*(FIX((FNa% 1)/4 FIX(0 75*
   (FIX((FNA% 1)/100)+1))))+(ABS
   (FNm%>2)*
   (-FIX((FNm%*0 4)+2 3)+FIX
   (FNa%/4)-FIX(0 76*(FIX(FNa%/
   100)+1))))
100 RESTORE 120
110 FOR X=0 TO 6:READ jour$(x):
    NEXT
120 DATA SAMEDI,DIMANCHE,LUNDI,
    MARDI,MERCREDI,JEUDI,
    VENDREDI
```

Bel algorithme, non ? Pour la saisie de la date, le format est jj-mm-aaaa.

COMPATIBILITE

Très intéressé par "Les mêmes programmes pour tous les CPC" de Daniel Le Barbier, CPC n° 21 (avril 87), Robert CABANE de PARIS 11 expose ses trucs pour obtenir GRAPHICS PAPER et GRAPHICS PEN.
Pour PEN, faire PLOT319,319,couleur. Cela fixe la couleur mais, pour l'affichage, il faut, au préalable, placer le curseur graphique au bon endroit. Pour PAPER, faire ORIGIN0,0,0,0,0 puis CLG 1, ce qui a pour effet d'afficher un petit trait en bas à gauche sur l'écran et met l'encre graphique au n° 1. Mais mieux vaut, sur 464, faire POKE &B338,PEN et POKE &B339,PAPER. Dans ce cas, on travaille sur un masque. Si l'on utilise des valeurs autres que 0, 1, 240 ou 255, on obtient des traces en pointillés simulant MASK.
Pour le PASCAL HI SOFT (concurrent de TURBO et tres rapide en graphisme), il est possible d'appeler les routines systèmes à l'aide de ce type de syntaxe ra.=c ; rhl =n ; user(# ,) ;

PHLYCTENES

Pour les inconditionnels d'ULTIMATE FIGHT (CPC n° 17), dont les doigts seraient sujets à des échauffements prononcés et dont le clavier se refuse à subir les derniers outrages, Michael HOFFERT, de THIONVILLE, propose les modifications suivantes

```
280 IF JOY(0)=1 etc.
290 IF JOY(1)=1
300 IF JOY(0)=2 ...
310 IF JOY(1)=2 ...
320 IF JOY(0)=8 ...
330 IF JOY(1)=4
340 IF JOY(0)<>8
350 IF JOY(1)<>4 ...
370 IF JOY(0)=8 ...
380 IF JOY(1)=4 ...
```

Désormais, c'est le joystick qui subira vos assauts !

APPEL A BIDOUILLE

Qui serait capable de réaliser un montage pour la conservation de la date du CP/M ?

Pierre TACONNET

Francis VERSCHEURE



Nous allons voir ce mois-ci, dans la suite de notre étude des fonctions du BDOS de CP/M, les fonctions de lecture et écriture en mode séquentiel, plus quelques fonctions diverses.

FIXATION DE L'ADRESSE DES TRANSFERTS DISQUE

Afin d expliciter le fonctionnement des fonctions de lecture et d'ecriture, il nous faut parler de la fonction 26 qui permet de fixer l'adresse des transferts disque

En effet, lors d une lecture séquentielle par exemple, CP/M va lire un secteur logique de 128 octets et le contenu de ce secteur sera mouvemente à l adresse courante des transferts disque Lors du démarrage de CP/M, cette adresse vaut 80H, mais la fonction 26 permet de fixer cette adresse de transfert ou vous la desirez, c'est a-dire le plus souvent dans une zone réservee a cet effet dans votre programme.

Si cela est necessaire cette adresse peut etre modifiee avant chaque lecture pour charger un fichier en mémoire par exemple ou pour changer de zone de transfert lorsque l'on passe d un fichier a un autre dans un programme qui travaille sur plusieurs fichiers

Bien sûr des que votre programme se termine, CP/M réinitialise cette adresse a 80H

En entree C = 26 DE = contient l adresse des transferts disque à fixer

En sortie pas de code retour

LECTURE SEQUENTIELLE D UN FICHIER

Cette fonction permet la lecture en sequentiel d'un fichier prealablement ouvert par la fonction 15. Rappelons que, pour CP/M, un fichier est un ensemble d enregistrements logiques de 128 octets (un "secteur" CP/M)

Donc, après la fonction 15 d'ouverture, le fichier est ouvert et le pointeur dans le fichier est fixé sur le premier enregistrement à lire

Chaque appel successif a la fonction 20 va donc lire 128 octets du fichier et les mouvementer à l'adresse fixee par la fonction 26 et déplacer le pointeur d'un enregistrement.

En entrée C = 20 DE = contient l adresse d'un FCB d'un fichier ouvert

En sortie A contient 0 si la lecture s est effectuée sans erreur

En CP/M 2 2 une valeur differente de 0 signale que l on a atteint la fin du fichier.

En CP/M Plus les différentes valeurs possibles de A sont

- 01 fin de fichier
- 09 . FCB invalide.
- 0A disquette a été changee
- FF erreur physique code dans H

ECRITURE SEQUENTIELLE D'UN FICHIER

Cette fonction permet d'effectuer des écritures sequentielles dans un fichier prealablement ouvert, soit par la fonction 15. soit par la fonction 22

Après l'ouverture le pointeur dans le fichier est fixé sur le premier enregistrement existant (fonction 15) ou à creer (fonction 22)

Chaque appel successif à la fonction 21 va donc prendre les 128 octets se trouvant à l adresse des transferts disque et les écrire dans le fichier concerne Si le fichier etait un fichier non vide et existant déjà les ecritures successives écrasent les anciennes donnees Si on

dépasse la fin du fichier tel qu'il existait à l'ouverture, celui-ci augmente en taille et la nouvelle fin de fichier est après le dernier enregistrement écrit
En entrée C = 21 DE = contient l'adresse du FCB d'un fichier ouvert
En sortie A contient le code retour, 0 si pas d'erreur et non nul dans le cas contraire
Sous CP/M Plus les codes possibles sont
— 01 plus de place dans le catalogue (en cas de création d'une nouvelle extension du fichier)
— 02 plus de place sur le disque pour les données
— 09 FCB invalide
— 0A disquette changee
— FF erreur physique (code dans H)

CREATION D'UN NOUVEAU FICHIER

Lorsque nous avons décrit la fonction 15, nous avons precisé qu'il s'agissait de l'ouverture d'un fichier déjà existant La fonction 22 permet de créer un nouveau fichier C'est à-dire qu'elle est identique à la fonction 15 sauf qu'il faut que le fichier spécifié soit inexistant dans le catalogue et que cette fonction cree le premier poste catalogue pour ce fichier

En entree C = 22 DE = adresse d'un FCB correctement initialisé
En sortie A contient une valeur de 0 à 3 si l'operation s'est effectuee correctement et 0FFH dans le cas contraire
Sous CP/M Plus le registre H contient le code erreur (voir fonction 15).
La figure 1 vous donne un exemple complet d'utilisation de ces fonctions
Ce programme recopie un fichier dans un autre, quelqu'en soit le contenu texte ou binaire, car la copie se fait par "secteurs" CP/M de 128 octets

CHANGEMENT DU NOM D'UN FICHIER

Cette fonction permet de changer le nom d'un fichier existant par un autre nom qui, lui, ne doit pas correspondre à un fichier dejà présent dans le catalogue La zone pointée par DE contient dans ses 16 premiers octets, le nom du fichier a renommer, dans le même format que pour un FCB
Les 16 octets suivants contiennent le nouveau nom du fichier également dans le format d'un nom de fichier de FCB mais avec le premier octet toujours à zéro
En entrée C = 23 DE = adresse d'une zone contenant l'ancien nom et le nouveau nom en format FCB
En sortie A contient un code retour équivalent à une ouverture de fichier, c est a dire de 0 à 3 en cas de succès et DFFH en cas d'erreur
Sous CP/M Plus en cas d'erreur le registre H contient le code de l erreur
La figure 2 vous donne un exemple d'utilisation de cette fonction

LECTURE DU VECTEUR D'ETAT DES DISQUES

Cette fonction permet de connaître les disques qui sont en ligne Un disque "en ligne" est un disque qui a éte accédé Ainsi dans un système à 2 lecteurs de disquette, si vous n avez pas effectué d'accès au lecteur B , celui-ci n'est pas en ligne Cette fonction ne permet donc pas de connaître le nombre de lecteurs maxi possibles dans un système
En entrée C = 24
En sortie le registre HL contient le vecteur des disques actifs
Le bit 0 de L correspond au disque A , le bit 1 au disque B jusqu au bit 7 de H qui correspond au disque P
Si le bit est a 0 le disque n'est pas en ligne
Si le bit est à 1, le disque est en ligne

LECTURE DU NUMERD DU DISDUE COURANT

Cette fonction permet de connaître quel est le disque courant, c est a-dire celui sur lequel on effectue les opérations disque lorsqu aucun disque n est specifie dans les noms de fichiers
En entree C = 25
En sortie A contient le numero du disque courant (A = 0 B = 1 ..)
Le mois prochain, nous continuerons avec quelques fonctions diverses mais surtout avec les lectures et ecritures en accès RANDOM (accès relatif).

FIGURE 1

```
;
; Ce programme lit un fichier et le recopie dans un autre.
;

BDOS     equ     5                  ; Appel du BDOS de CP/M.

CONOUT   equ     2                  ; Affiche un caractere.
PRINTS   equ     9                  ; Affichage d'un message.
OPEN     equ     15                 ; Ouverture d'un fichier existant.
CLOSE    equ     16                 ; Fermeture d'un fichier ouvert.
CREATE   equ     22                 ; Creation d'un fichier.
DELETE   equ     19                 ; Suppression fichier existant.
READ     equ     20                 ; Lecture sequentielle.
WRITE    equ     21                 ; ecriture sequentielle.
SETDMA   equ     2E                 ; Fixe adresse des transferts disque.

CR       equ     0DH                ; Retour chariot.
LF       equ     0AH                ; A la ligne.


; Debut du programme

         lxi     d,MSGDEB           ; Affiche message de debut.
         mvi     c,PRINTS
         call    BDOS

; Ouverture du fichier a copier.

         mvi     c,OPEN             ; Ouverture du fichier
```

```
        lxi     d,FCBIN         ;  a recopier.
        call    BDOS
        inr     a
        jz      Erreur1         ; Fichier non trouve.
        mvi     c,DELETE        ; Suppression du fichier
        lxi     d,FCBOUT        ;  en sortie si existe deja.
        call    BDOS
        mvi     c,CREATE        ; Et creation d'un nouveau
        lxi     d,FCBOUT        ;  fichier en sortie.
        call    BDOS
        inr     a
        jz      Erreur2         ; Erreur de creation fichier en sortie.
        lxi     d,BUFFER        ; adresse du tampon pour les operations disque.
        mvi     c,SETDMA
        call    BDOS

; Boucle de lecture du fichier en entree

Lit:    mvi     c,READ          ; Lecture d'un enregistrement en entree.
        lxi     d,FCBIN
        call    BDOS
        ora     a
        jnz     FinDeFichier    ; Fin de fichier ou erreur grave.
        mvi     c,WRITE         ; Ecriture d'un enregistrement en sortie.
        lxi     d,FCBOUT
        call    BDOS
        ora     a
        jnz     ErreurEcriture  ; Erreur a l'ecriture.
        mvi     c,CONOUT        ; Affiche un point
        mvi     e,'.'           ;   pour signaler que
        call    BDOS            ;     le programme travaille.
        jmp     Lit             ; sinon reboucle sur lecture.

FinDeFichier:
        cpi     1               ; Fin de Fichier?
        jnz     Erreur4         ; NON affiche l'erreur.
Fermetures:
        lxi     d,FCBIN         ; Fermeture du fichier en entree.
        mvi     c,CLOSE
        call    BDOS
        lxi     d,FCBOUT        ; Fermeture du fichier en sortie.
        mvi     c,CLOSE
        call    BDOS
        inr     a
        jz      Erreur2         ; si erreur a la fermeture.
        jmp     Fin

Erreur1: lxi    d,MSGERR1       ; Message d'erreur numero 1
Erreur: mvi     c,PRINTS        ; Affiche message
        call    BDOS            ;   adresse dans DE.
        jmp     Fin             ;    et termine.

Erreur2: lxi    d,MSGERR2       ; Message numero 2
        jmp     Erreur

ErreurEcriture:
        lxi     d,MSGERR3       ; Message numero 3
MsgEtFerme:
        mvi     c,PRINTS        ; Affiche
        call    BDOS
        jmp     Fermetures      ; et ferme les fichiers.

Erreur4: lxi    d,MSGERR4       ; Message numero 4
        jmp     MsgEtFerme

; Fin du programme
```

```
Fin:    lxi     d,MSGFIN            ; Affiche message de Fin
        mvi     c,PRINTS
        call    BDOS
        mvi     c,0
        call    BDOS

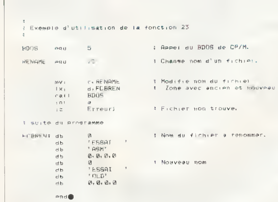
MSGERR1: db     CR,LF,'Erreur ouverture fichier en entree.$'
MSGERR2: db     CR,LF,'Erreur creation fichier en sortie.$
MSGERR3: db     CR,LF,'Erreur ecriture sur fichier en sortie.$'
MSGERR4: db     CR,LF,'Erreur lecture sur fichier en entree.$'

MSGDEB: db      CR,LF,LF,'Debut du programme',CR,LF,LF,'$'
MSGFIN: db      CR,LF,LF,'Fin du programme',CR,LF,LF,'$'

FCBIN:  db      0                   ; FCB du fichier en entree.
        db      'ESSAI   '
        db      'ASM'
        db      0
        ds      36-($-FCBIN)

FCBOUT: db      0                   ; FCB du fichier en sortie.
        db      'ESSAI   '
        db      'SAV'
        db      0
        ds      36-($-FCBOUT)

        db      'Buffer='
BUFFER: ds      128
        db      'Fin buffer'
```



```
;
; Exemple d'utilisation de la fonction 23
;

BDOS    equ     5                   ; Appel du BDOS de CP/M.

RENAME  equ     23                  ; Change nom d'un fichier.

        mvi     c,RENAME            ; Modifie nom du fichier
        lxi     d,FCBREN            ;  Zone avec ancien et nouveau noms.
        call    BDOS
        inr     a
        jz      Erreur1             ; Fichier non trouve.

; suite du programme

FCBREN: db      0                   ; Nom du fichier a renommer.
        db      'ESSAI   '
        db      'ASM'
        db      0,0,0,0
        db      0                   ; Nouveau nom
        db      'ESSAI   '
        db      'OLD'
        db      0,0,0,0

        end
```

FIGURE 2

RESTAURATION DE LA MEMOIRE BASIC

Bernard BESSE

Le manuel du CPC 464 dit ceci à propos de HIMEM :
"Avant de redéfinir l'octet le plus haut avec la commande MEMORY, il est souhaitable d'écrire mm = HIMEM. Vous pourrez après retourner à la capacité de mémoire précédente avec la commande MEMORY m."
Le manuel mentionne donc la question de la restauration de la mémoire et c'est bien.
Malheureusement, le problème est d'une part mal posé, d'autre part mal résolu.

Nous proposons une procédure tout à fait différente de restauration mémoire. Mais, auparavant, nous devons répondre aux 3 questions suivantes :

— Est-il utile de restaurer la mémoire (autrement que par RESET) ?
— En quoi le problème est-il mal posé dans le manuel ?
— Pourquoi est-il mal résolu ?

FAUT-IL RESTAURER LA MEMOIRE ?

Sauf en cas de RESET, tout programme "hérite" de l'organisation mémoire léguée par son prédécesseur. Cette organisation a pu, éventuellement, être modifiée par des commandes en mode direct. Cette "passation de suite" se fait évidemment pour un même programme d'un RUN au RUN suivant.

L'erreur bien connue "improper argument" intervenant au 2ᵉ RUN d'un programme commençant par :

SYMBOL AFTER nnn
MEMORY zzzz

tient au fait que le 2ᵉ RUN hérite des zones SYMBOL et MEMORY créées par le 1ᵉʳ et que le BASIC refuse alors de créer une 2ᵉ zone SYMBOL.

La memoire BASIC peut être réduite par la réservation des zones suivantes :

SYMBOL AFTER (0 à 1 zone)
MEMORY (0 à 3 zones)
FICHIER (0 à 1 zone)

Nota : s'il existe une zone FICHIER, il existe au moins une zone MEMORY, car nous supposons une pratique sans faille de :

OPENOUT"bidon" MEMORY HIMEM 1
CLOSEOUT

Voici un exemple de configuration à 5 zones qui peut surprendre, mais qui est tout à fait possible.

Il faut, pour la réaliser, commencer par faire SYMBOL AFTER 256 car il ne peut y avoir qu'une seule zone SYMBOL AFTER, comme nous l'avons vu.

BASIC	BAS
Memory C	(souvent 1 octet)
Fichier	(4096 octets)
Memory B	(B octets)
Symbol	(8 à 2048 octets)
Memory A	(A octets)
Ram Systeme	HAUT

Il est probable que les petits programmes n'ayant pas de besoins particuliers pourront s'accommoder d'un tel héritage. Mais pour :

— les gros programmes,
— ceux qui ont besoin d'une réservation d'espace à une adresse précise,
— ceux qui ont besoin d'une zone SYMBOL AFTER plus importante,

il ne peut être question de conserver l'organisation léguée par le programme précédent.

Par ailleurs, la solution bien connue qui consiste à faire RESET pour réinitialiser la mémoire n'est pas sans inconvénients :

— effacement des instructions KEY,
— effacement du programme en cours,
— pas de possibilité d'écrire RUN et LOAD dans un programme pour appeler le programme suivant.

On peut donc affirmer qu'il est très souhaitable de pouvoir restaurer la mémoire autrement que par RESET.

EN QUOI LE PROBLEME EST-IL MAL POSE ?

Il est mal posé parce qu'il est plus logique de restaurer en DEBUT qu'en FIN de PROGRAMME.

En effet, pour un programme donné, la restauration au début donne la certitude d'avoir une organisation mémoire convenable.

Dans l'autre cas (restauration en fin), cette certitude ne peut s'obtenir qu'en programmant la restauration mémoire dans TOUS les programmes susceptibles de précéder le programme concerné.

De plus, il faudrait s'interdire l'utilisation des ordres MEMORY, SYMBOL AFTER et OPEN en mode direct.

Pour illustrer le caractère sournois d'un oubli, voici un petit exemple :

Supposons qu'un programme hérite d'une zone MEMORY 30000 et qu'il ait besoin d'une zone MEMORY 40000.

Le programmeur qui suit le Manuel écrira :

mm = HIMEM	' donc mm = 30000
MEMORY 40000	' la mémoire monte à 40000 (sans problèmes supposons le)
—	
MEMORY mm	' la mémoire baisse à 30000 valeur ' léguée au programme suivant. Bravo !

LA PROCEDURE "mm = HIMEM \ MEMORYmm" DU MANUEL NE MARCHE PAS

Procédons à un certain nombre d'essais dans les conditions ci-dessous :

a) Par hypothèse, les instructions suivan-

(1) mm HIMEM MEMORY HIMEM 1000 MEMORY mm OUI

(2) SYMBOL AFTER 128 mm = HIMEM MEMORY HIMEM 1000
MEMORY mm NON

Note HIMEM fin HIMEM debut 1024

(3) mm HIMEM SYMBOL AFTER 128 MEMORY HIMEM 1000
MEMORY mm NON

Note Memory full (en MEMORY mm)

(4) mm HIMEM OPENOUT bidon MEMORY HIMEM 1 CLOSEOUT
MEMORY mm NON

Note Memory full (en Memory mm)

(5) mm = HIMEM MEMORY HIMEM 1000
n 1000 DIM a$(n)
FOR k = 1 TO n a$(k) = "" + MEMORY OPEN SYMBOL NEXT k
h1 TIME MEMORY mm
PRINT USING ####.## (TIME h1) 300

Note entre MEMORY mm et la fin s'écoulent 3m 42s ou en théorie non en pratique nécessaires à la réorganisation des chaînes à la nouvelle adresse supérieure du BASIC

tes font partie de chaque essai
— RESET et SYMBOL AFTER 256 avant chaque essai
— PRINT HIMEM au tout début et à l'extrême fin de l'essai
b) Les caractères \\ délimitent les instructions centrales (non écrites si elles sont sans intérêt)

c) Les résultats sont notés
OUI si HIMEM fin = HIMEM deb
NON si ERREUR ou si HIMEM différents

Conclusion des essais

Le procédé du manuel est à rejeter pour les raisons suivantes
— MEMORY ne peut pas récupérer une zone SYMBOL AFTER (seul SYMBOL AFTER 256 peut le faire)
— MEMORY ne peut pas récupérer une zone FICHIER OUVERTE (seul CLOSEOUT et/ou CLOSEIN peuvent le faire),
— MEMORY ne peut récupérer une zone FICHIER FERMEE (du genre OPENOUT"bidon") qu'a une seule condition
Au moment ou l'ordre MEMORY va être exécuté, la valeur de HIMEM doit être celle créée par l'ordre OPENOUT (avant le MEMORY HIMEM-1)
C'est ainsi que l'essai n° 4 réussit si l'on remplace MEMORY mm par MEMORY HIMEM + 1: MEMORY mm

UNE AUTRE SOLUTION

La solution du problème réside dans l'utilisation de 2 adresses gérées par le BASIC.
Ces 2 adresses se trouvent en AE7B/C (il s'agit tout simplement de HIMEM) et le couple d'octets voisin en AE7D/E.
Nous appellerons MEMOIRE LIBRE (ou plutôt mémoire libérable par MEMORY) et en abrégé MMLIB cette 2e adresse

Utilisation de MMLIB et HIMEM

MMLIB est la valeur maximum du paramètre que l'on peut associer à MEMORY.
Nous avons toujours MMLIB > = HIMEM
En conséquence
Si HIMEM < MMLIB on doit faire MEMORY mmlib
Si HIMEM = mmlib, MEMORY ne doit plus être utilisé (car l'ordre MEMORY HIMEM n'a aucun effet)

Nous avons alors 2 cas de figure théoriquement possibles

a) Nous sommes 1 octet plus bas qu'une zone FICHIER OUVERT

b) Nous sommes 1 octet plus bas qu'une zone SYMBOL AFTER

Puisque nous restaurons en début de programme, le cas a est éliminé car aucun fichier n'est ouvert en début de programme
Reste le cas b que l'on traite par SYMBOL AFTER 256
Ceci se traduit par les 2 lignes de BASIC ci après

```
100 mmlib = PEEK(&AE7D) + 256*
    PEEK(&AE7E) IF HIMEM < mmlib
    THEN MEMORY mmlib GOTO 100
200 hiold = HIMEM SYMBOL AFTER
    256: IF HIMEM > hiold THEN 100
```

On pourrait, pour arrêter le processus, comparer HIMEM à la valeur MAXI de 42647 (ou 44031 pour les cassettes), au lieu de vérifier que sa valeur reste égale à HIOLD
Malheureusement, l'utilisation de certaine(s) routine(s) du système d'exploitation a pour résultat apparent de diminuer le MAXI en le rendant égal à la valeur du HIMEM courant
C'est en effet la regrettable comportement de la routine &BB4E utilisée inconsidérément
Ainsi, le petit programme
RESET
SYMBOL AFTER 258

```
100 PRINT HIMEM      42747 à la 1ère
                     itération
100 '                puis 42619 puis
                     42491 etc
110 SYMBOL AFTER 240
120 CALL &BB4E
130 SYMBOL AFTER 256 ne fait rien
                     (HIMEM non
                     augmenté)
140 GOTO 100
```

grignotera la mémoire BASIC à raison de 128 octets à chaque itération avec "memory full" pour HIMEM = 16507
L'explication peut être considérée comme la suivante
Call &BB4E met à zéro l'octet &B295
Cet octet a la valeur 255 lorsqu'une matrice de caractères est définie, la valeur 0 après un SYMBOL AFTER 256.
Par ailleurs, l'octet B294 contient le numéro du 1er caractère "utilisateur".
Par exemple, on aura
— Après SYMBOL AFTER 240
B294 = 240 et B295 = 255
— Après SYMBOL AFTER 0
B294 = 0 et B295 = 255
— Après SYMBOL AFTER 256
B295 = 0 et B294 = 0
On voit que la valeur de B295 permet de distinguer SYMBOL AFTER 256 et SYMBOL AFTER 0 qui sont notés tous deux zéro en &B294
Call &BB4E fait en quelque sorte "croire" au système qu'on vient de faire un SYMBOL AFTER 256 alors qu'il n'en est rien
Sur cette base frauduleuse, le système considère alors que MAXI = HIMEM
Nous n'avons pas trouvé un moyen sûr de contrecarrer les agissements délétères de &BB4E (que cette routine soit appelée à partir du BASIC ou d'une routine machine)
Il y a donc lieu, soit de s'en tenir aux 2 lignes de BASIC précédentes soit d'ajouter une 3e ligne pour "prévenir". On a alors

```
100 mmlib = PEEK(&AE7D) + 256*PEEK
    (&AE7E) IF HIMEM < mmlib THEN
    MEMORY mmlib GOTO 100
200 hiold = HIMEM SYMBOL AFTER
    256 IF HIMEM > HIOLD THEN 100
300 MAXI = 42747 IF HIMEM < maxi
    THEN PRINT "HIMEM", HIMEM,
    "<", MAXI," Faire RESET"
```

La valeur de MAXI est bien entendu à remplacer par 44031 pour les systèmes à cassette
Il est souhaitable de stocker ces 3 lignes sur disque (ou sur cassette) sous format ASCII
Elles pourront être exécutées par RUN ou intégrées par MERGE aux programmes qui ont besoin d'une restauration mémoire
Notons que MMLIB, HIOLD sont des variables réelles. Il faudra donc fusionner ces 2 ou 3 lignes avant un ordre DEFINT éventuel

Les articles suivants, parus dans CPC, précisent certaines questions
— Le bug de l'openout (Michel Archambault) n° 3, p 48
— Fichiers (Michel Archambault) n° 18, p 22
— Symbol After (Michel Archambault) n° 12, p 44
— Test (Patrick Guyon) n° 19, p 89